An Ontology-based Semantic Web for Arabic Question Answering: The Case of E-Government Services

نظام السؤال والجواب القائم على قاعدة الأنطولوجي والويب الدلالي: حالة
الخدمات الحكومية الإلكترونية

by

ALI BAHA’EDDIN ALBARGHOTHI

Dissertation submitted in fulfilment

of the requirements for the degree of

MSc INFORMATICS
(KNOWLEDGE AND DATA MANAGEMENT)

at

The British University in Dubai

August 2018

DECLARATION

I warrant that the content of this research is the direct result of my own work and that any use made in it of published or unpublished copyright material falls within the limits permitted by international copyright conventions.

I understand that a copy of my research will be deposited in the University Library for permanent retention.

I hereby agree that the material mentioned above for which I am author and copyright holder may be copied and distributed by The British University in Dubai for the purposes of research, private study or education and that The British University in Dubai may recover from purchasers the costs incurred in such copying and distribution, where appropriate.

I understand that The British University in Dubai may make a digital copy available in the institutional repository.

I understand that I may apply to the University to retain the right to withhold or to restrict access to my thesis for a period which shall not normally exceed four calendar years from the congregation at which the degree is conferred, the length of the period to be specified in the application, together with the precise reasons for making that application.

Signature of the student

COPYRIGHT AND INFORMATION TO USERS

Abstract

Since the beginning of the digital age, the amount of information has increased significantly, and the relationships among the different types of information have become more sophisticated. The fact that the number of diverse users is growing urges researchers to benefit from this information and develop techniques that analyse customers' experiences to meet their needs. As far as e-government is concerned, the number of structured and unstructured webpages of electronic services has increased, making the repositories more complex and harder to analyse without considering semantic knowledge. Related studies have highlighted some challenges in the Arabic Semantic Web (SW) that adversely impact their results, for instance, data heterogeneity, and the differences in ontology construction approach. In this study, we present an approach for automatic extraction of an ontology-based SW constructed from Arabic webpages related to Dubai's e-government services. Furthermore, we use the constructed ontology as the knowledge base for a question-answering (QA) task process to answer questions related to e-government services. The proposed methodology consists of two stages. The first stage is automatic ontology construction for Dubai government services. This stage is concerned with data extraction and validation from the Dubai government portal[1] that includes the official profiles for more than 500 services. After that, the Natural Language Processing (NLP) tasks are used to process the services' profiles and extract the ontological keywords. Next, we map the rules to link the ontology components with the extracted keywords. Lastly, the ontology is constructed using the OWL format. In the second stage, an Arabic QA approach is implemented to answer user questions relevant to e-government services. This stage comprises three steps: question analysis, information retrieval (IR), and answer validation. We conducted experimental performance evaluation for all stages in our methodology. The ontology construction stage reported high scores in terms of precision, with 87% on average, and recall, with 97% on average. Further, 414 automatic questions are tested on the QA algorithm using two methods, semantics-based and keyword-based. The accuracy results show 90% for semantics-based and 72% for keyword-based. These results confirm that the semantics-based approach significantly outperforms the keyword-based approach.

[1] http://www.dubai.ae

Keywords: Ontology Construction; Natural Language Processing; Keyword Extraction; OWL; Government Services; Protégé; Arabic; Question Answering, Semantic Web.

ملخص

منذ بداية العصر الرقمي، ازداد حجم المعلومات بشكل ملحوظ، وأصبحت العلاقة بين الأنواع المختلفة للمعلومات أكثر تعقيدًا. إن حقيقة تنوع عدد المستخدمين يحث الباحثين للاستفادة من هذه المعلومات وتطوير تقنيات لتحليل تجارب العملاء لتلبية احتياجاتهم. وفيما يتعلق بالحكومة الإلكترونية، فقد ازداد عدد صفحات الويب الإلكترونية المنظمة وغير المنظمة، مما يجعل مصادر المعلومات أكثر تعقيدًا وصعوبة في التحليل دون النظر إلى المعرفة الدلالية. وقد إشارات الدراسات السابقة إلى بعض التحديات في الويب الدلالي العربي والتي تؤثر سلبا على نتائجها، مثل عدم تجانس البيانات والاختلاف في أسلوب بناء الأنطولوجيا. في هذه الدراسة، نقدم أسلوبا للاستخلاص التلقائي للويب الدلالي القائم على الأنطولوجي والذي تم بناؤه من صفحات الويب العربية المرتبطة بخدمات حكومة دبي الإلكترونية. بالإضافة إلى ذلك، يتم استخدام الأنطولوجيا المبنية كقاعدة معرفية لنظام السؤال والجواب للإجابة على أسئلة المستخدمين المتعلقة بخدمات الحكومة الإلكترونية. تتكون المنهجية المقترحة من مرحلتين. تهتم المرحلة الأولى بالإنشاء التلقائي لخدمات حكومة دبي. وتعتمد هذ المرحلة استخراج والتحقق من بيانات أكثر من 500 خدمة موجودة في بوابة دبي الإلكترونية. بعد ذلك، يتم استخدام مهام البرمجة اللغوية العصبية لمعالجة بيانات الخدمات من تحديات اللغة العربية واستخراج الكلمات المفتاحية. بعد ذلك، نقوم بتعيين القواعد لربط مكونات الأنطولوجي مع الكلمات الرئيسية المستخرجة. أخيرا، يتم بناء الأنطولوجي باستخدام لغة أنطولوجي الويب. في المرحلة الثانية، يتم تطبيق أسلوب الإجابة والسؤال باللغة العربية للإجابة على أسئلة المستخدم ذات الصلة في خدمات الحكومة الإلكترونية. تحتوي هذه المرحلة على ثلاث خطوات: تحليل السؤال، استرجاع المعلومات والتحقق من الإجابة. تم إجراء التقييم التجريبي لكافة مراحل المنهجية. وتشير مرحلة بناء الأنطولوجي إلى درجة عالية من حيث الدقة بنسبة 87 ٪ في المتوسط ، والاستعادة بنسبة 97 ٪ في المتوسط. أيضا، تم فحص نتائج 414 سؤال آلي على نظام السؤال والجواب باستخدام طريقتين، تعتمد على قاعدة الويب الدلالي وقاعدة الكلمات المفتاحية. تظهر دقة النتائج 90% لأسلوب قاعدة الويب الدلالي ونسبة 72% لأسلوب قاعدة الكلمات المفتاحية. تؤكد هذه النتائج أن الأسلوب القائم على الويب الدلالي يتفوق بشكل كبير على أسلوب الكلمات المفتاحية.

Dedication

I dedicate this to my parents, to my lovely wife, to my kind children, my brothers, my sisters and my whole family. I really appreciate and thank them for patience, support through study period. Further, I dedicate this to my friends who are provided me full support in this dissertation. I also dedicate my dissertation supervisor who guided me in this process and motivate me to keep me on track.

Acknowledgement

Thanks to Allah Almighty for guidance and support to complete this dissertation. The work presented in this dissertation would not have been possible without the help, advice, inspiration, and encouragement of many people.

I am grateful to Prof. Khaled Shaalan for guidance, motivation and moral support. He always helped me out when I got any difficulties or queries regarding my study.

I would like to thanks the domain experts who supported my study by providing an assistance in constructing services dataset and validating the questions answers.

Many Thanks to my family, my parents, my sisters and my brothers, my colleagues for all things you do for me, your pray, patience, motivation and continues support.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviations	**Description**
IR	Information Retrieval
SW	Semantic Web
NLP	Natural Language Processing
QA	Question Answering
QAS	Question Answering System
NL	Natural Language
SPARQL	SPARQL Protocol and RDF Query Language
OWL	Web Ontology Language
RDF	Resource Description Framework
XML	Extensible Markup Language
W3C	World Wide Web Consortium
HTML	Hypertext Markup Language
WWW	World Wide Web
POS	Post of Speech
SVM	Support Vector Machine
TF/IDF	Term Frequency / Inverted Document Frequency
NER	Named Entity Recognition
IRI	Internationalized Resource Identifier
AWN	Arabic WordNet

1. Chapter One: Introduction

In the digital evolution era, information is the major pillar of the internet as it contains a large number of documents, images, videos, newspapers, etc. Digital content makes it easy to explore and retrieve all sorts of information. Many applications and search engines such as Google, Bing, and Yahoo use information retrieval techniques (Rosso, Benajiba & Lyhyaoui 2006). IR returns the documents relevant to a user's query from a large collection of pages or documents within a few seconds (Mislove et al. 2007).

These search engines retrieve their answers as webpages or links. Therefore, the exact answer is not retrieved directly. This is attributable to the huge amount of information, which is being updated continuously. What's more, search engines perform their tasks well by retrieving the pages that contain the correct answers, but there is a good deal of information that is retrieved incorrectly. Question answering systems (QASs) provide the optimal solution to this issue by asking the question in NL and manipulating the question using natural language processing to analyse it and retrieve the correct answer. These systems can help many users to obtain the correct answers and reduce their search times. Furthermore, companies' customers need trusted systems with which to inquire about services and procedures and receive accurate answers instead of calling the call centre and wasting their time on the phone or other communication channels. This chapter will point out the importance of QASs for all users and how the government's customers can benefit from QASs (Gorenjak, Ferme & Ojsteršek 2011).

1.1. Background

1.1.1. Questions through Search Engines

Most users depend on search engines to get answers to their inquiries through the retrieved pages or documents. For instance, when Google was established in 1998, thousands of users used the site for their searches daily. After one year, Google was able to retrieve and answer more than 3.5 million queries per day. Currently, Google handles more than 40,000 users' requests every second. Figure 1.1 presents the search query statistics history between 1999 and 2012 for the Google search engine (*Google Search Statistics – Internet Live Stats* n.d.).

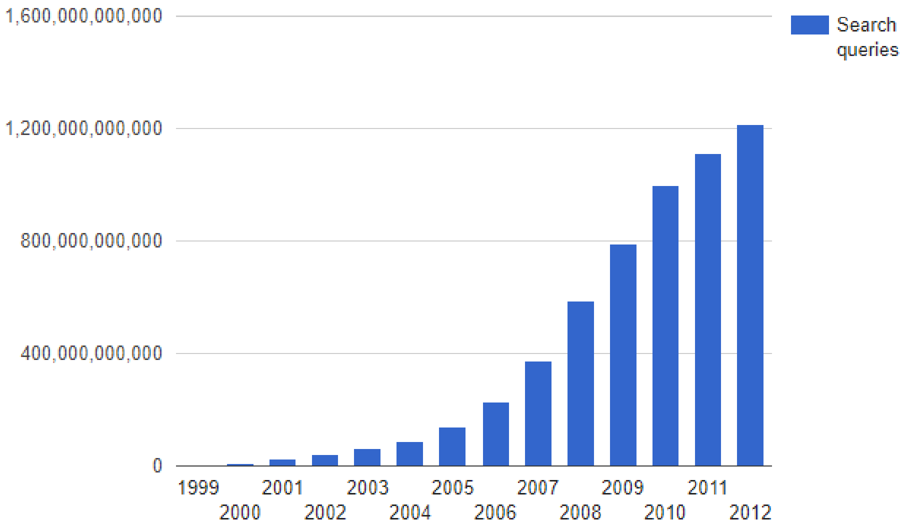


Figure 1.1 Google Search Queries Statistics

However, there are many other search engines available, such as Yahoo, Ask.com, and Microsoft Bing. Figure 1.2 shows the comparison between them by calculating the number of core search queries for the best search engines in the United States between 2008 and 2017. The number of queries in the below graph is in billions. Google is the leading search engine in the United States, processing around 10 billion queries monthly, whilst other search engines have processed less than 3 billion queries per month. In total, 66% of search queries have been handled by Google (Statista 2018).

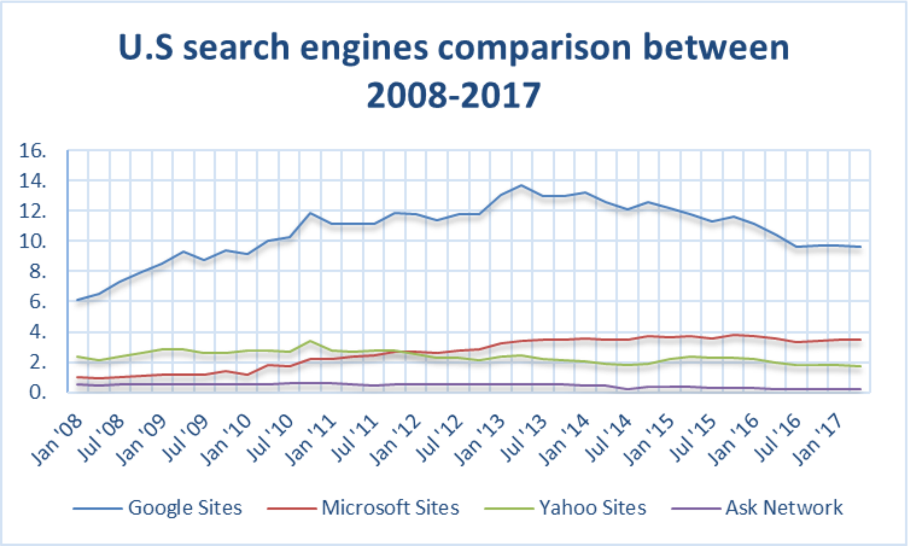


Figure 1.2: Search Engines Comparison

1.1.2. Question-Answering Systems (QASs)

Users are looking for the right answer by asking a question, but the answer exists in multiple sources of information. In order to retrieve the correct answer, QASs help users obtain the right answer after extracting and validating the potential answers from various sources (Benajiba et al. 2014). The idea of a QA system is asking questions using natural language and analysing the question through NLP by conducting linguistic analysis, which helps to understand the user query and link the question with an appropriate source of information (AbuTaha & Alagha 2015). In addition, Ezzeldin and Shaheen (2012) said that the objective of QASs is to analyse the question and the sources of information (documents, databases, etc.) by using an NLP approach to match the questions' output with candidate documents that have answers. NLP has a lot of functions to manipulate text and documents, including tokenization, stemming, named entity recognition (NER), and part of speech (POS) tagging . Noy et al. (2001) explained the objective of QASs as returning the correct answers derived from a massive data source (either structured or unstructured) to particular questions. The author clarifies that a structured data source is the optimal way to return an accurate answer. Therefore, QASs use an ontology- or knowledge-based data source, since in such a source there is an obvious domain or concept with clear attributes or slots. Ray and Shaalan (2016) pointed out several types of QASs, including online and offline systems. These systems have a common architecture, with some differences in domain, knowledge base, IR, and language support.

Systems and approaches to the question-and-answer task are various, so having a generic design and architecture is important. Systems with specific selections can characterize the question-and-answer architecture to represent and handle the general model parts. Figure 1.3 shows the three main headings (question processing, answer processing, and document processing), which are divided into further subheadings.

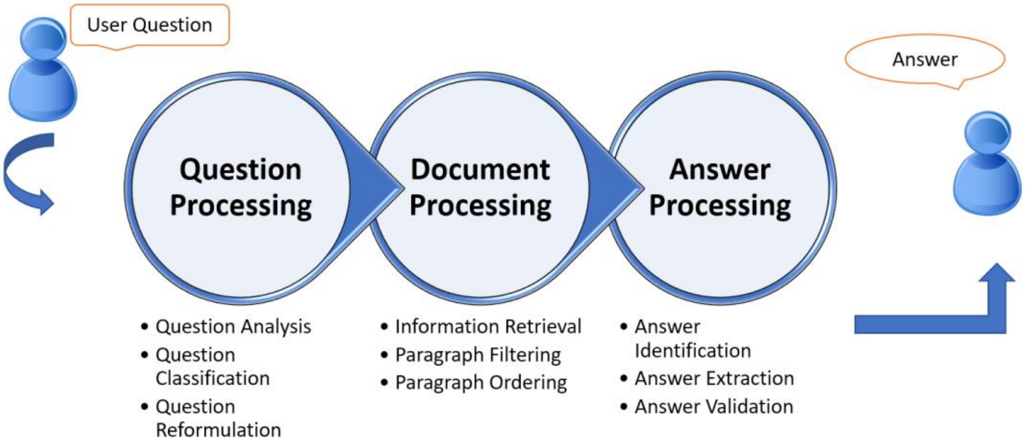


Figure 1.3: QA Components

First, question processing deals with the classification of questions. Second, document processing carries out information retrieval. Third, answer processing is responsible for answer extraction (Allam & Haggag 2012).

As for question processing, it takes the user's question in NL as an input, analyses it, and links the meaning of the user's question with the query in the system. Other steps include classifying the type of the question and formulating it as a yes/no question or other similar question (Gupta & Gupta 2012). Next, according to Stoyanchev, Song and Lahti (2008), document processing reformulates the pattern, then IR recall finds the possible answer, although if the answer is not available, the user will get no response. Besides, the accuracy of IR determines the accuracy of the answer. If no right answers are available in the documents, the system will not be able to retrieve the answer. IR accuracy and ranking depend on the selected documents, which will also affect QA performance. Finally, answer processing extracts the previously prepared answer to the question from a given paragraph (Lampert 2004). According to Allam and Haggag (2012), the user asks the NL question, then the process moves to the question section to analyse, classify, and formulate it and pass it to document processing. After reaching document processing, the question passes three phases. First, the paragraph is reordered. Second, with the help of the IR system, the paragraph is filtered. Finally, the information needed is retrieved. Along the document process, an independent IR system is used to collect data from various document corpora (Lampert 2004). Allam and Haggag (2012) stated that the question's journey from question to answer passes from question processing to the IR system that collects and lists the related documents. Document processing ranks the retrieved documents, filters the ranked documents, and sorts the required paragraphs. Finally, answer processing is responsible for identifying, extracting, and validating answers from the list of selected paragraphs. Accordingly, answer processing deals with answers by selecting words or phrases related to the question and checking their accuracy.

1.1.3. Semantic Web

After the advent of the SW concept to organize information as a structured data source, the SW became a good choice for knowledge representation and reasoning. Ontologies represent the main concepts, interactions, and classes among a set of description logic (DL) in the real world. Besides, ontologies are quite appropriate for arithmetic analysis of users. But the structures of ontologies could be understood in a weakly insightful form, for example, classes, relationships, and properties. This also applies to users who have more experience in the ontology domain when they analyse conceptual data or update the knowledge data source. Therefore, people may wish to obtain more information regarding NL in ontology (Gyawali 2011). Jain and Singh (2013) have pointed out the need for the SW to enhance the knowledge network by building structure for information and enriching the web repository. In 1996, Sir Tim Berners-Lee created the idea of the SW to change the shape of information into a reasonable and understandable form.

1.1.4. E-Government Concepts and E-Services

With a high-tech revolution taking place around the world, most governments depend on information and technologies to perform their daily work and implement their plans. Many studies have been conducted on e-government to develop e-government concepts and build their own model (Gil-Garcia & Martinez-Moyano 2007). Bekkers (2003) mentioned that the public sector cares about communication and information in order to improve government infrastructure and services by investing in information technology (IT) and to transform the government’s capabilities to meet customers’ needs and expectations. Gil-Garcia and Martinez-Moyano (2007) presented the stages of e-government, which help to identify the benefits of e-government on the technological and organizational levels. According to the authors, the stages are initial, extended, interactive, transactional, vertical integration, horizontal integration, and totally integrated. Each one has its own business model, infrastructure, users, technology, and services.

Furthermore, in the United Arab Emirates (UAE), the government's direction is to automate all public services by adopting the latest technologies and delivering smart services using mobile technology. This critical stage started when His Highness Shaikh Mohammad bin Rashid Al Maktoum, Vice-President and Prime Minister of the UAE and Ruler of Dubai, announced the conversion from e-government to smart government and the establishment of a new approach to work in government (Khaleej Times 2013). What's more, Dubai's government puts more emphasis on this approach by launching many initiatives to adopt smart services within government entities, such as the happiness meter to monitor customer satisfaction and encourage government entities to provide their services in a smarter way. To achieve this objective, government agencies consider their customers as partners by getting their inputs, feedback and satisfaction level for each provided service (GulfNews 2015).

1.2. Importance of Constructing an Ontology-based SW to E-Government Services

E-government is considered as the civil conduct that uses information and technologies continuously. The government's information is distributed within multiple systems and technologies, which impacts information integration and extraction. In addition, the government's policies and regulations encounter huge challenges in reaching integration and interoperability (Alazemi, Al-Shehab & Alhakem 2017). Furthermore, the aim of e-government is to provide citizens with easy access to services and enough information through effective service browsing and execution (Gil-Garcia & Martinez-Moyano 2007). Therefore, integration between e-government services is required to achieve this objective.

E-government services are considered the main body of knowledge and should be organized and available for all citizens. Charalabidis and Metaxiotis (2009) pointed out that the main knowledge classifications within e-government, public administration as well as systems and applications meet citizens' needs through service interaction. The application framework of knowledge management within e-government is divided into three phases: (1) **PUBLISH,** in which IT procedures are used to gain access to e-government information without reaching government offices, avoiding the long queue. (2) **INTERACT,** whose aim is to expand civic engagement in government activities, for example, meetings between citizens and government officials to discuss multiple issues and building a government/citizens forum. (3) **TRANSACT,** whose aim is to automate all government processes to make them available online to all citizens. For example, automating the processes of tax collection could eliminate corruption and increase trust in government. Moreover, these procedures could improve productivity in the government and the private sector.

In order to implement the SW, the government agencies face many challenges. For instance, data heterogeneity which is related to service information exchange. Another challenge is the technology middleware. In order to technology middleware in e-government, the applications must comply with the government's regulations and policies (Guijarro 2009). Additionally, applying the SW in e-government will help the government to capture and share knowledge (resources, services, etc.). In addition, the SW will help the e-government to automatically configure the specifications to customers' needs (Gil-Garcia & Martinez-Moyano 2007).

Alazemi, Al-Shehab and Alhakem (2017) presented two types of ontology construction for services: Manually and automatically developing. The process begins with extracting the main terms and concepts from services' data sources. Figure 1.4 illustrates the semantic service for an OWL file.

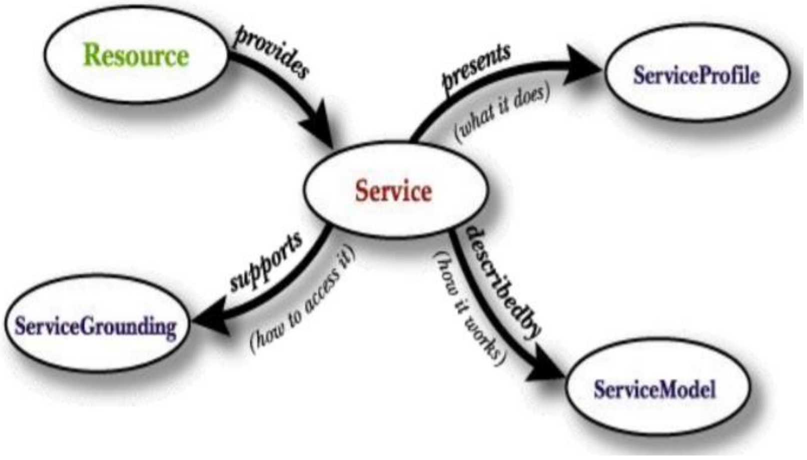


Figure 1.4: Semantic Service in OWL File

As a result, a valid method for implementing an ontology for e-government services should start with constructing and integrating the core services of government by using the formal ontological definitions as the basis of knowledge indexing and reusing (Charalabidis & Metaxiotis 2009).

1.3. Usefulness of QA Systems to Customers of E-Government Services

1.3.1. Problem Definition

Usually, the services provided by the government are complicated and have a lot of requirements to complete the service request. At the same time, the services’ consumers have a lot of inquiries answered by domain experts. In addition, consumers must review and understand the services’ description to retrieve some of the required information on their own. Therefore, implementing online services is necessary for all customers. QAS can answer direct questions related to e-government services instead of retrieving full text documents (Schwarzer et al. 2016).

Furthermore, several studies have focused on the SW for particular English domains, while the Arabic ontology studies are limited and address specific domains. Arabic e-government services ontology studies have not been widely conducted because of challenges in the Arabic language and information availability. Moreover, Arabic question answering tools for SW studies are still bounded and have not achieved the standard set in English. This dissertation aims to resolve this problem by building an Arabic ontology for e-government services and applying QAS.

1.3.2. Research Objectives

The main objective of this dissertation is to automatically construct an Arabic ontology for e-government services. In addition, this dissertation aims to answer Arabic customers’ questions by developing a QA algorithm and using a SPARQL queries to translate NL questions into a SW structure to derive the answers from the ontology. There are other sub-objectives for this research:

1. To conduct a literature review of current studies to get a solid background before building the methodological framework.
2. To analyse current e-government services in order to build the ontological structure and components.
3. To develop a tool to extract services’ information from Arabic webpages.
4. To build an approach to automatically constructing the ontology in order to save effort in the design phase.

5. To extract the keywords and terms from the ontology and build the rules according to it.
6. To perform NLP tasks to resolve Arabic challenges and organize the ontological components.
7. To evaluate the constructed ontology.
8. To apply QA to SW.
9. To build baseline questions from the ontology.
10. To translate NL questions into a SPARQL queries.
11. To evaluate the QA algorithm with the Google search engine.

1.3.3. Research Motivation

As mentioned earlier, the amount of information has increased significantly, and the relations between the sources of information have become more complicated. In addition, the number of users has grown to benefit from this information, and the types of users vary, which requires analysing the customers' questions to meet their needs in this regard. Within the UAE, Dubai has developed itself remarkably in the last 20 years. The city provides more than 500 government services to its inhabitants, who consist of both expats and citizens. Dubai's government decided to adopt smart services and increase customers' satisfaction; therefore, the need to provide effective and higher-quality responses to customers' queries is high. The main motivation for this research is that information is available online on Dubai's government's website. This means it is possible to build a dataset for government services and build a QAS based on that. Consequently, the Dubai government will give more attention to its services to be sustainable, electronic, smart, and simpler by developing various methodologies, for example, smart services, artificial intelligence, and paperless communications. In this dissertation, the author focuses on Dubai's government services by studying the implemented services' profiles for each government entity. In addition, the research will come up with a QAS for service information, which will play a main role in increasing customers' satisfaction. Also, this approach may lead to better realizing clients' needs regarding service information whilst reviewing the services' profiles and evaluating the answers to questions about all services. Furthermore, this study builds a QAS based on ontology and exploring Arabic requirements for supporting QAS and helping Arabic users to retrieve the right answers to their questions.

1.3.4. Research Questions

The dissertation's aim is to solve the defined problem and diagnose it properly through achieving the research objectives. Therefore, we identified the below research questions that emanate from the problem definition and research objective.

1) **Is it possible to build a QA system that can answer government-related questions as accurately as possible, deal with Arabic QA challenges, and reach acceptable performance?**

   This question is related to the main objective "To answer Arabic customers' questions by developing a QA algorithm and using SPARQL queries". In addition, it is linked to sub-objectives 8, 9, 10, and 11.

2) **How can we construct an ontology that is related to e-government services and enable information retrieval in a structured ontology?**

   This question is related to the main objectives "To answer Arabic customers' questions by developing a QA algorithm and using SPARQL queries" and "To automatically construct an Arabic ontology for e-government services". The related sub-objectives are 1–11.

3) **How can we enhance the QAs based on a rule-based approach?**

   This question is related to the main objective "To answer Arabic customers' questions by developing a QA algorithm and using SPARQL queries". The related sub-objectives are 8, 10, and 11.

1.3.5. Research Methodology

To build a solid methodology, research on public sector services has been performed to measure the importance of government services to their customers and to understand the government's directions regarding transforming the services from manual/electronic into smart. This type of analysis gives us a holistic understanding of government services' maturity in the public sector, which leads to understanding the current challenges and using a clear approach to build a solid dataset using the SW. This dissertation pursues an experimental methodology and is divided into two phases: firstly, automatic ontology construction for Dubai government services; secondly, building a QAS using NLP techniques and SPARQL queries, then applying this system to the constructed ontology. The methodology follows the right track to finding the right answers to the research questions.

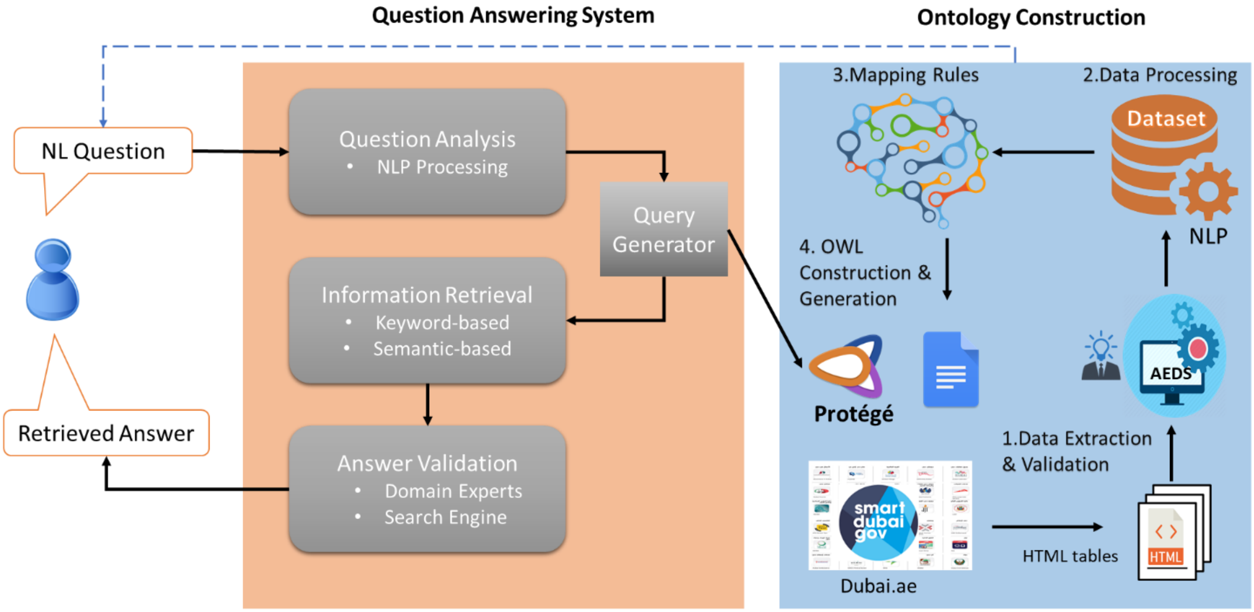


Figure 1.5: Methodology Framework

Figure 1.5 presents the high-level representation of our methodology. The automatic construction of the ontology phase has the following steps: data extraction and validation, data processing, mapping rules, and web ontology language (OWL) generation. Meanwhile, the Arabic QAS phase steps are preparing baseline questions, question analysis, IR and ontology mapping, and answer validation.

1.4. Innovation of the Present Study

The dissertation consists of seven chapters, which are built as follows:

- **Chapter 1** introduces the research background about search engines, QA, the SW, and e-government services. Also, the problem statement, research motivations, objectives, research questions, and methodology are mentioned in this chapter.
- **Chapter 2** presents a literature review of SW concepts, tools, and components.
- **Chapter 3** presents a literature review of QA concepts, architecture, and approaches. In addition, Arabic NLP and its challenges are highlighted.

- **Chapter 4** presents the methodology of automatically constructing an ontology from Arabic webpages and the detailed steps of each part, supported with examples. In addition, the experimental results for this phase are discussed.
- **Chapter 5** presents the methodology of the QAS and the steps of translating NL into SPARQL queries using two methods (semantics-based and keyword-based). The results and comparisons are highlighted and discussed in this chapter.
- **Chapter 6** answers the research questions.
- **Chapter 7** reveals the study's conclusion and limitations and recommendations for future works.

In summary, we presented a brief introduction to common search engines, which showed with numbers the importance of information to users. Further, we presented the QAS's objectives and the main types of QASs. Additionally, the SW concept was mentioned, and its importance was explained to enhance knowledge and information representation. Moreover, we showed the importance of information and technologies for e-government as well as the benefits of SW to enhance customer service. The UAE was used as an example of a government that emphasises service automation to improve service efficiency and sustain its business. Finally, the problem statement, research objectives, research motivation, methodology, and dissertation structure were presented in this part to show the importance of this dissertation.

2. Chapter Two: Semantic Web Literature Review

2.1. Overview

The SW will fulfil the future need to enhance the present web, make it more efficient by constructing a consistent data structure, and increase its size. In 1996, Sir Tim Berners-Lee imagined that a machine could convert information into a reasonable form. This idea was crystalized into the SW (Jain & Singh 2013). Furthermore, semantic search is used to resolve the keyword-based search method's weakness in answering a question correctly due to different meanings of the identified concepts. The SW understands the aims of users' questions or searches and the actual meaning of the class or concept within the query. SW usage has increased search performance by adding intent and concept analysis for each term in the query text (Waller 2016).

SW technology has provided good results with languages that use the Latin alphabet, and there is a great opportunity to use it for the Arabic language as well (Al-Zoghby & Shaalan 2015b; Ray & Shaalan 2016). SW technology aims to organize information into a structured data source. In addition, the SW is a good choice for knowledge representation and reasoning about government services (Gyawali 2011). Jain and Singh (2013) pointed out the need for the SW to enhance the knowledge network by building a structure for information and enriching the web repository. Albarghothi, Khater and Shaalan (2017) stated that the ontology is considered as a knowledge representation of concepts in the domain as well as the relations between these concepts.

2.2. SW Definitions

The SW characterises the data that are used by a machine or system for automation, reuse by the software, and integration (Laboratory for Data Technologies n.d.). The SW and the WWW transform unstructured data into a knowledge representation. This representation becomes a platform that allows the represented data to be available through applications and to be reused (Jain & Singh 2013).

Waller (2016) defined *semantics* as a "meaning or relevant meaning in language". The SW is illustrated as a set of activities that provide meaning or understanding to the web or machine.

Berners-Lee and Fischetti (2001) indicated that the meaning of any information or terms in web content may be observed not only by human tasks but also by computer systems. In addition, Berners-Lee and Fischetti (2001) believe that SW may augment human knowledge by adding meaning to web content. The SW concept is defined from different directions: The World Wide Web Consortium (2012) considered the SW as linking web content data with machines such that a computer can automate, integrate, and reuse them in several data applications or knowledge data sources.

2.3. SW Architecture

Sir Tim Berners-Lee, the inventor of the World Wide Web and the manager of the World Wide Web Consortium (W3C), explained the architecture of the SW as being in the shape of a SW stack. This visualizes the linguistic hierarchy, in which each layer can utilize the features of the below layers. In addition, it presents how the SW technologies are organized to develop the SW and make it applicable (W3C 2015). Figure 2.1 illustrates the parts of the SW stack that form three layers: (1) Hypertext web technologies: This is in the bottom layer and includes well-known technologies. The main technologies used are the Uniform Resource Identifier (URI), which supplies a unique ID for SW resources, allowing provable updates in resources in the upper layers, as well as Unicode, which helps to represent the text in multiple languages. It also includes Extensible Markup Language (XML), which formulates the text resources in structured data. XML namespaces are required to connect the data and refer to resources in a single document. (2) Standardized SW technologies: These are in the middle layer and serve to develop SW applications by using the following technologies: the Resource Description Framework (RDF), which is used to represent the concepts in graph mode formed from triples; the RDF Schema (RDFS), which consists of the major words for RDF that create the hierarchies among classes and attributes; OWL, which explains the semantics of RDF by adding more constraints; SPARQL, a query language that returns the answer from SW applications; Rule Interchange Format (RIF), a crucial tool that permits description of the relationships that are not described by DL in OWL. (3) Unrealized SW technologies are in the top layer and contain information to realize SW. The technologies are cryptography, which is used to verify SW statements and ensure that they are received from a trusted layer; trust, which helps ensure that all derived statements are supported and come from a trusted source; and the user interface, the last layer that permits the user to use SW applications.

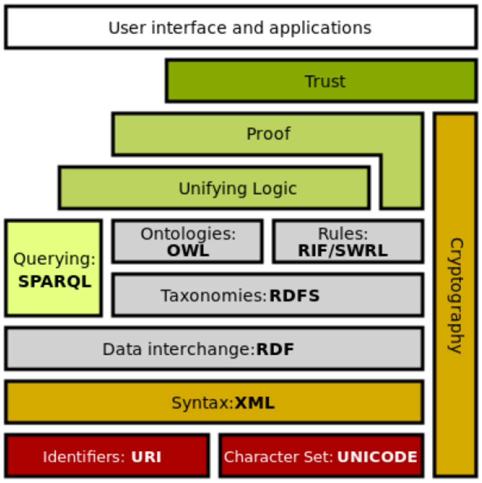


Figure 2.1: Semantic Web Stack

Jain and Singh (2013) represented the SW architecture according to specific syntax and guidelines. XML allows writing and building structured documents. RDF is a data model that enables building expressions for web resources. In addition, RDF and RDFS supply hierarchies for objects according to modelling primitives. Moreover, the basic primitives are defined as a domain, classes, subclasses, properties, relations, and ranges. The logic layer produces XML/RDF documents by developing the ontology, which is called knowledgeable representation. The next part is a proof layer, which contains the deductive process, proof validation, and languages. Lastly, the trust layer is created during the utilization of the digital signatures and other knowledge.

According to Lim and Sun (2005), the main difference between the web and the SW is that the WWW has content for human consumption. In addition, the web's structure has formatting instructions that are used in its presentation, whereas the SW contains the main data for tools' use.

The main purpose of using RDF is to represent data by using data model forms that contain triples (subject, predicate, and object). Meanwhile, RDFS is considered the base of the RDF layer that maintains the vocabulary and represents the RDF data model. XML consists of the structure of the data model on the web. The last layer is Unicode and the URI. Unicode enacts each character purely and has own intellectual style, whilst URI represents information in a data model (Yadav et al. 2016).

2.4. Semantic Web Technologies

According to Hitzler, Krotzsch and Rudolph (2009), the SW has been considered as an extension of the WWW that permits machines to search according to meaning. Without using artificial intelligence, this cannot be done if the meaning of the web source is not specified clearly so it can be processed by computers. For that reason, the authors claimed that it is important to use the data source semantically instead of storing data in an HTMP page. Therefore, to cope with this requirement, semantic technologies have evolved (OWL, SPARQL, RDF, etc.). These technologies help users to save information on the web, identify the words, and create the rules for manipulating data.

2.4.1. RDF

RDF is used to formulate and represent knowledge based in structured data. Its main objective is to apply the vision of the SW in which all web resources are annotated with semantic tags, as well as to make them clear and easy to use for computers and machines.

Ordinarily, RDF is presented in a graph that contains a group of subjects or objects that are connected by predicates. This connection between nodes and edges is called URI. Figure 2.2 shows an example of RDF containing two nodes and one edge (Hitzler, Krotzsch & Rudolph 2009).

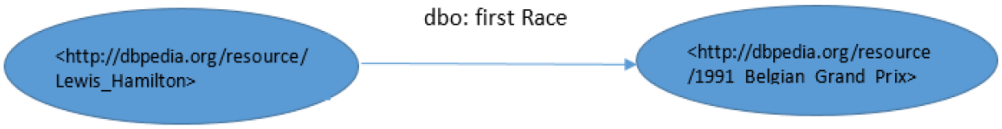

<http://dbpedia.org/resource/
Lewis_Hamilton>
dbo: first Race
<http://dbpedia.org/resource
/1991 Belgian Grand Prix>

Figure 2.2: RDF Graph Example

The dbo abbreviation is labelled with “first Race” and points to the address “http://dbpedia.org/ontology/”. Both nodes are labelled with URIs to distinguish them from each other. “Literals” are the data values reserved in RDF resources with datatype. Also, the list of characters presents the value for each literal. Figure 2.3 shows an example of a literal with data value description. The left node represents the literal of the driver’s name, while the box represents the data value of the driver, “Lewis Hamilton”. The RDF triple has three parts: The subject represents the resource being described, “dbr:Lewis Hamilton”; URI is the representation of the subject. The object has the value of the resource in the relation “Lewis Hamilton”, whereas the predicate mentions “dbp:name” (Mehdipour Pirbazari 2017).

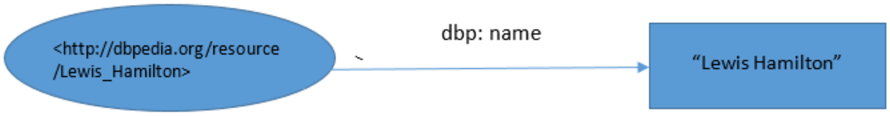


Figure 2.3: Literal RDF graph example with data value

2.4.2. RDFS

Ontology is required to form the data semantically by adding the identity and the structure to the current data via URIs and RDF. As mentioned before, the ontology contains the knowledge base constructed from the set of concepts with shared vocabulary and each of which has properties and rules. RDFS is considered the main structure to identify the ontology of RDF real data. It permits the definition of classes/properties with hierarchies, along with the range and domain of each attribute. The element rdfs:class is responsible for defining the class, whereas the element rdf:type is used to define the instances. In addition, rdf:Property is another element to define the properties, which have some restrictions defined by rdfs:range and rdfs:domain. Moreover, the relationship is created by sub/super-classes through rdfs:subClassOf and rdfs:superClassOf (Mehdipour Pirbazari 2017).

2.4.3. OWL

As discussed in Section 2.3, W3C has the OWL standard for ontology language, which is constructed from RDF and RDFS and supplies more vocabulary for classes’ and properties’ definitions (Patel-Schneider & Horrocks 2006).

2.4.4. SPARQL

SPARQL is considered as an RDF query language for databases able to manipulate and retrieve data from an ontology that has RDF format (Hebeler et al. 2009). Gorenjak, Ferme and Ojsteršek (2011) defined SPARQL as a standard query similar to SQL that uses keywords (Select, Where, Group by, etc.), although SPARQL uses additional keywords that do not exist in SQL (Optional, Filter, etc.). McCarthy, Vandervalk and Wilkinson (2012) pointed out that SPARQL can deal with and adapt RDF. However, the query syntax is complicated for most users, even those with experience. Therefore, new techniques have been suggested to support SPARQL building and construction. Ou et al. (2008) stated that SPARQL is built with three patterns (subject, predicate, and object) and finished with a full stop. Each pattern has variables that accept the values of subjects and objects.

2.5. Ontology Components

Noy et al. (2001) defined the ontology concept as a group of concepts, objects, and classes standing in a known area of relations and benefits. Further, ontology is defined as a clear, hierarchical vision that provides a logical outcome. Besides, ontology comprises a knowledge base by joining a group of classes, concepts, relations, objects, slots, restrictions, and instances that are linked with classes.

Sosnovsky and Dicheva (2010) stated that ontology has two types: heavyweight and lightweight. Lightweight ontology consists of taxonomy or class hierarchy that has classes, attributes, and values. On the other hand, heavyweight ontology contains many more details, for example, constraints and axioms.

Additionally, Noy et al. (2001) described ontology's components as constraints, relationships, and concepts that could be represented through ontology layers. The authors presented the advantages of using ontology in the SW: The first benefit is sharing knowledge and information among users and tools. Secondly, ontological knowledge is reusable by representing the concepts and their relationships. Thirdly, determining the impact of changes in knowledge scope makes it easier to develop hypotheses that make it explicit. Fourthly, ontology permits multiple users to present their own domains in the ontology, which evidences a collaborative system. Finally, Rubin, Noy and Musen (2007) described ontology as an essential part of software to convert data into binary code using ontological concepts.

2.6. Ontology Development

Noy et al. (2001) provided the primary motives for constructing ontologies and listed the stages of ontology development. Within the sequence of constructing the ontology, there are some principal guidelines that have to considered in the ontology's layout: First, to envision a specific area, there may not be one approach; however, there is usually an opportunity to use techniques. Second, ontology layout and development is an iterative method. Ultimately, the object and the relationships within the ontology must be near the idea of ontology inside the decided-on domain.

There are steps to build any ontology: (1) Decide on the required ontology domain design by answering questions related to the domain, such as the scope of the ontology, the ontology users, and the purpose of using ontology. (2) The chosen domain can be reused or expanded and might need integration with different applications associated with the managed words or precise ontology (Boyce & Pahl 2007). (3) After that, prepare the major terms/keywords within the selected ontology that will help determine the words list that can be used. (4) Class or concept hierarchy identification is useful to organize the list of terms in various methods, for instance, top-down, middle-out, and bottom-up. (5) The class attributes or properties (slots) are used to describe ontology concepts (Gilmmour 2004). (6) After identifying the concepts and slots, the types and constraints will describe, for example, slot data type, slot allowed values, slot constraint, etc. (7) The final product after completing the previous steps represents the knowledge base of ontology. Figure 2.4 summarizes the seven steps of ontology development.

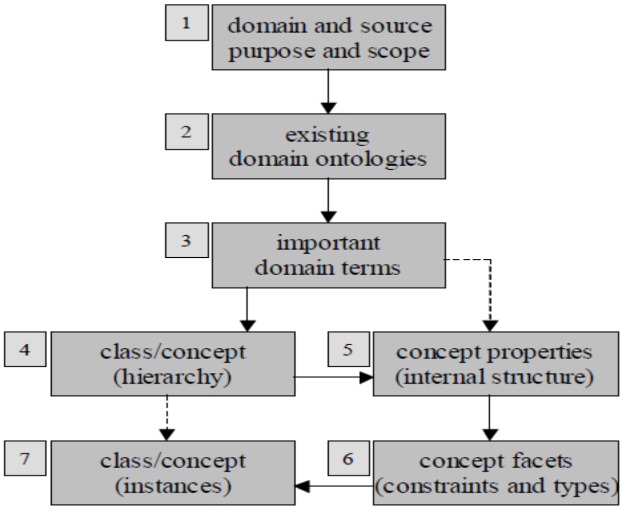

1
domain and source purpose and scope
2
existing domain ontologies
3
important domain terms
4
class/concept (hierarchy)
5
concept properties (internal structure)
7
class/concept (instances)
6
concept facets (constraints and types)

Figure 2.4: Ontology Development Steps

2.7. Ontology Evaluation

The produced ontology has a lot of complicated relations and terms. To ensure the quality of the ontology, an evaluation should be conducted based on a set of criteria to show the ontology’s richness, complexity, and granularity. Brank, Grobelnik and Mladenić (2005) pointed out a set of techniques for ontology evaluation according to ontology types and objectives. Ontology evaluation is classified into four sorts (golden standards, task-based, data-driven, and experts) (Obrst et al. 2007).

Dellschaft and Staab (2006) clarified golden standards using multi-dimensional evaluation, which helps the users to evaluate multiple types of errors according to their preferences. Moreover, the strengths and weaknesses of learned ontology can be analysed better. Another criterion is the gap between the correct instance and the result. Clarke et al. (2013) summarised a task-based approach as steps using a group of annotations to identify terms. Furthermore, an enrichment analysis is required to know the annotations' efficiency and retrieve an accurate result. After this quality assessment is performed, the annotations' completeness, precision, and accuracy are measured.

Shah, Shah and Deulkar (2015) explained that a data-driven approach is used to assess collected data source information with ontology. This approach shows how to understand the ontological references to a specific subject and classify them into concepts, relations, and slots. Brank, Grobelnik and Mladenić (2005) claimed that assessment by humans requires manual efforts to assess the ontology according to clear criteria; the expert should tag and check each concept and object in the ontology and record all types of errors. In addition, the expert has to ensure all ontology levels (concepts, taxonomy, relations, context, syntax, and architecture) are evaluated well.

2.8. Ontology Tools

OWL is used as an SW language to gather knowledge on groups and the interactions among them. The W3C is a community that contains many members from various organizations unified to create web standards. OWL is a component from W3C that uses technologies such as SPARQL and RDF to develop an ontology (World Wide Web Consortium 2012).

OWL has three types (Lite, Description Logic, and Full). OWL Lite helps those users who frequently need a class hierarchy and easy constraints. In addition, it helps with implementation when OWL is used. OWL Description Logic (DL) was created to aid users who require maximum expressiveness whilst maintaining computational completeness. OWL Full was created to aid users who require maximum expressiveness without the syntactic restrictions of RDF and without computational completeness (World Wide Web Consortium 2012).

Hamdy and Shaalan (2018) mentioned that when building an ontology using OWL, it is necessary to add a layer that facilitates semantic integration with other SW layers. Yadav et al. (2016) stated that information in the OWL form produces ontologies and can be saved in one document and published to the WWW. The content of the ontology contains four main axioms (header, class, property, and individual). The class has a mechanism to group a set of properties. For instance, the RDF class has a group of individuals called “Class Extension”. Therefore, the classes may have the properties of class extension. In addition, each class has its own description, Uniform Resource Identifier (URI) reference, property restriction, and relations.

Protégé[2] is one of the well-known ontology tools and an open-source platform. This tool is used to create the knowledge bases for specific domains. In addition, it produces a group of class hierarchies that represent knowledge visually and can maintain an ontology in several formats. Moreover, Protégé can be easily customized for the ontological domain by creating knowledge concepts and feeding it with data. In addition, Protégé can integrate with other SW platforms. Generally, the Protégé-OWL API is applied for RDF and web ontology; the API provides methods and concepts to generate OWL syntax files. Furthermore, the API has a query and update feature to maintain the data models. DL Query is a powerful and simple approach for searching within a classified ontology; it gathers specific information for class, object, or relation in a single frame (*Protege Wiki* 2010).

2.9. Related Work

Recently, efforts in Arabic ontology building have increased. Most of these efforts have handled Arabic ontologies with NLP techniques: for instance, IR (Al-Zoghby & Shaalan 2015a), text annotation (Hazman, El-Beltagy & Rafea 2012), text summarization (Imam et al. 2013), and QA (Abouenour, Bouzoubaa & Rosso 2008; Al-Chalabi, Ray & Shaalan 2015). The Arabic language has impressive power that makes it more complicated to automatically construct ontologies. Consequently, the efficient automatic extraction of these relations is a complex

[2] http://protege.stanford.edu/

approach that relies on dictionaries (Jarrar 2010). This section presents the related work of different SW tools and frameworks that is built on multiple methodologies to achieve high-performance results.

2.9.1. Ontology Extraction from Text Documents

Benabdallah, Abderrahim and Abderrahim (2017) proposed a novel approach to constructing ontologies from Arabic text resources according to semantic relations and keywords/terms. The authors used an Arabic dictionary for synonyms and antonyms and Arabic WordNet, a lexical database that is rich with semantics. In addition, the authors used learning linguistic markers to connect the extracted terms and link them using semantic relationships. For quality purposes, domain experts manually tagged the correctly extracted entities and reported good results, with 89% precision and 82% recall. On the other hand, Haarmann, Gottsmann and Schade (2012) presented a framework that explains how to build ontologies for WIKI texts using an information extraction (IE) system and text mining. This domain was prepared by energy research experts with a glossary format. The IE system organizes the SW contents and syntactic data in WIKI texts. A text mining approach was used to automatically extract an ontology from the text.

2.9.2. Building Ontologies from Textual Webpages

Al-Safadi, Al-Badrani and Al-Junidey (2011) highlighted that the morphological analysis of traditional Arabic is a challenge in the Arabic SW and does not produce good results. Consequently, a modern analysis of Arabic text was adopted with their equivalence in Traditional Arabic to improve performance. For example, "هاردوير"is parallel class of (Hardware) “مكون مادي”. The experimental results were poor and showed 50% for precision after running SPARQL queries in the Jena[3] API format.

2.9.3. Building Ontologies from XML Sources

Ferdinand, Zirpins and Trastour (2004) proposed a method to build an ontology using the OWL format based on XML schemata and transform it into RDF graphs. The mapping process between XML and OWL files was based on mapping rules. In addition, the classes and data objects

[3] http://jena.sourceforge.net/

emerged from XML schemata and their elements. Xu and Li (2007) proposed a method to construct ontologies from XML documents using the entity-relation model. This model uses a process called XML Transform to Relational Database (XTR) for an XML document and then makes a Relational Database Transform to Ontology (RTO) in order to map an entity to an OWL ontology.

2.10. Summary

In this literature review, we presented the main background of the SW. The SW and ontology were explained, and the needs for the SW to enhance web search and structure emerged. In addition, SW definitions and architecture were discussed, as well as the main SW technologies, for example RDF, OWL, and SPARQL. Moreover, we presented the main ontology tools used, such as Protégé.

This literature review makes a major contribution to answering parts of research questions 2 and 3. Most of the studies illustrated manual approaches to constructing an ontology, which consume effort and time. Since most e-government services are published online, we need a solution to answer the customers' questions that are relevant to the services. Therefore, we decided to propose a joint approach to automatically construct ontologies from webpages, using ontology keywords to bridge the rule gaps between the main entities' terms and ontology components.

3. Chapter Three: Question Answering Literature Review

The objective of this literature review is to prepare a comprehensive view on QA that provides in-depth understanding of NLP and QA. In addition, it aims to answer the research questions by highlighting the approaches and challenges in these subjects to build a solid framework and a holistic view of QA.

3.1. Natural Language Processing

Being easier for people to use, computer interfaces are of high interest to computer science researchers. It will not be necessary for people to learn a specialized programming language to deal with a computer. Instead, they will use their own language. One of the most interesting fields of artificial intelligence and language among researchers is NLP (Mihalcea, Liu & Lieberman 2006). The main concern of NLP is the difficulty of interaction between human language and computers in terms of theory, application, and information sharing. NLP techniques fall into a broad range, including morphological analysis, translation, information retrieval, text categorization, and dictionary automation. In addition, there are two open sources of NLP system software, the Stanford parser and Open NLP (Gangwal 2012).

3.2. Arabic Language Challenges and Word Analysis

3.2.1. Arabic Language Challenges

Arabic is spoken by 300 million people, making it the sixth–most common language in the world. Additionally, Arabic is the official language in 22 Arab countries and the language of Islamic instruction. As for the features of the Arabic language, it is written from right to the left and consists of 28 letters. Arabic roots, which mostly consist of three constants, are the source of most Arabic words (Al-Shalabi et al. 2009). Comparatively, the Arabic language is different from other languages, such as English. The English language has benefitted from the wide-ranging research in this area, whereas the research on Arabic is still in its initial stages. There are some factors that have held the Arabic language back in this area, including the following (Hammo, Abu-Salem & Lytinen 2002):

- Diacritics are not found in the modern Arabic language writing system, which makes text difficult to interpret and necessitates a lot of complex rules to identify characters and analyse texts.
- The writing direction goes from right to left, and the shapes of some characters change according to their position in a word.
- Capitalization is not used in Arabic, which makes it difficult to differentiate between common and proper nouns and abbreviations.
- Morphological analysis becomes too complicated, as Arabic is highly inflected and derivational.

On the other hand, technically, the Arabic language lacks machine-readable dictionaries which are fundamental to advancement in this field.

The extensive increase in Arabic digital content on the World Wide Web and the dramatic increase in the number of Arabic users of the internet make it important for Arabic language processing tools to process this content and interact with Arabic-speaking users. The complexity of the Arabic language's morphological structure necessitates morphological analysis as the first step. Having its own syntactic tags such as infixes, prefixes, and suffixes makes it important to have syntactic analysis. This means the output of morphological analysis is used in higher steps of Arabic processing, such as POS tagging and syntactic analysis (Sonbol, Ghneim & Desouki 2009).

3.2.2. Arabic Word Analysis

Word analysis in Arabic can be classified as stem-based or root-based. Stem-based means removing all the suffixes and prefixes without touching the infixes, while root-based depends mainly on returning the word to its root (Al Ameed et al. 2005). Khoja (2001) produced a stemmer that removes the affixes, then compares the remainder of a word to the pattern to excerpt the root, and finally checks the root using an Arabic roots dictionary. Grammatically, Arabic words are classified as nouns (اسم), verbs (فعل), and particles (حرف). Figure 3.1 shows the major part of speech categories (POS).

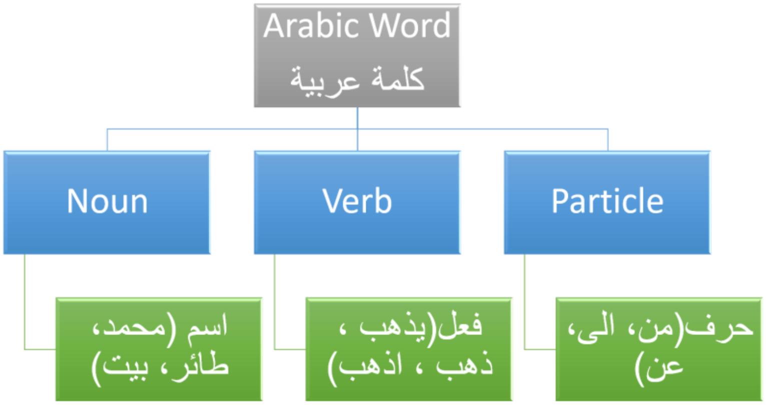


Figure 3.1: Arabic Word (POS)

Verbs are words that express actions and states; verbs can, for example, indicate the present (فعل مضارع) and past tenses (فعل ماضي) and the imperative mood (فعل أمر), for example in Arabic, يذهب، ذهب، اذهب, respectively. Meanwhile, nouns are words that refer to things such as people (اسم), ideas (صفه), and places (مكان). Finally, particles can be adverbs, conjunctions, prepositions, or interjections (of, to, in, on, for, oops etc.); some examples in Arabic include من، إلى، عن (Al-Shalabi et al. 2009).

In addition, Benajiba, Rosso and Lyhyaoui (2007) claimed that Arabic has complicated morphology, i.e., inflectional and derivational sentences. For instance, derivation uses the following base:" Lemma=Root+Pattern", as shown in Figure 3.2.

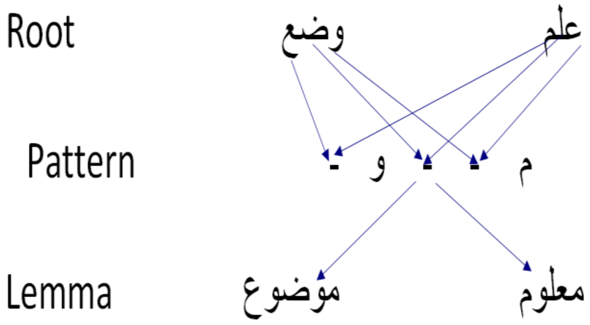


Figure 3.2: Derivation Rule

An Arabic word contains a root and affixes which can express multiple meanings through the inflectional process. Affixes include suffixes, prefixes, and infixes, each of which can be added to a word in order to construct a new meaning, as presented in below example in Figure 3.3. The root word is “مسـك” (“catch”), with the added the affixes “فسـي”, “ون”, “ها”, producing the word “فسيمسكونها” (“and they will catch it”).

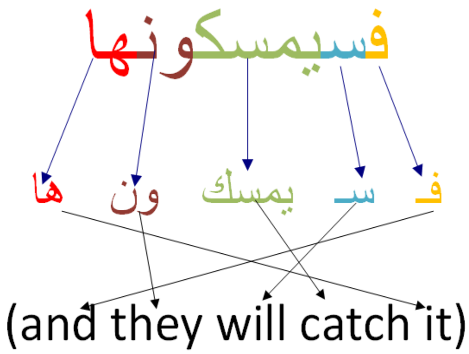


Figure 3.3: Inflectional Example

3.3. Arabic Question Answering

Arabic studies on QA are still limited and have not reached the maturity of those in English QA; this limitation is caused by some challenges in the Arabic language. Moreover, there were some recent Arabic QA implementations using unstructured knowledge bases, although the results did not reach the level of other languages (Kurdi, Alkhaider & Alfaifi 2014). Under those circumstances, there has been some research performed on Arabic-language challenges, proposing some solutions to resolve these issues and challenges. For example, inflection and derivation in Arabic words lead to thinness in NLP analysis.

3.3.1. Arabic QA Systems

Arabic QASs take less chance than those in other languages, which is attributable to language challenges, semantic style, language structure, and limited research (Kurdi, Alkhaider & Alfaifi 2014). In this part, Arabic QASs will be presented, and the purpose of each will be discussed. Mohammed, Nasser and Harb (1993) proposed the first Arabic QA system, called a knowledge-based Arabic QA **(AQSA).** AQSA was developed according to a knowledge-based method that can return the answer to the asked question from a structured database. The concept of this system is analysing the Arabic declarative question based on query-phrase and interpreting the text to produce the knowledge. AQSA has five components: first, a parser to split the declarative sentence into tokens and perform some morphological analysis; second, a dictionary, which contains the content word categories of Arabic (open, closed); second, a knowledge base, used as a frame that contains the source, effects, and application; fourth, an interpreter, whose main purpose is to retrieve the right answer and store it in an internal representation; and finally, the generator, which is responsible for showing the answers to the user after returning the results from the interpreter. Mohammed et al. (1993) claimed some challenges related to morphology analysis are caused by using a strict dictionary. Hammo, Abu-Salem and Lytinen (2002) described the design and approach of Arabic QA called the **QARAB** system**.** QARAB accepts a question in Arabic and returns a short answer. The system uses two techniques, IR and NLP, to build the knowledge source from text documents. The main phases of this system are document processing, question processing, and answer processing.

Benajiba, Rosso and Lyhyaoui (2007) proposed a new architecture of AQA. Also, they implemented an Answer Extraction (AE) module for factoid questions by using NER and passage retrieval modules. The value of adding an AE module is that it increases the results' accuracy. The system achieved 83.3% precision for factoid questions. Below, Figure 3.4 describes the architecture of this system.

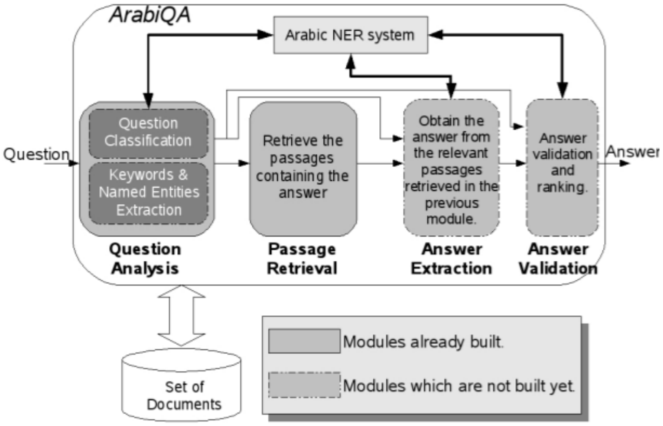

ArabiQA
Arabic NER system
Question Classification
Keywords & Named Entities Extraction
Retrieve the passages containing the answer
Obtain the answer from the relevant passages retrieved in the previous module.
Answer validation and ranking.
Question
Answer
Question Analysis
Passage Retrieval
Answer Extraction
Answer Validation
Set of Documents
Modules already built.
Modules which are not built yet.

Figure 3.4: ArabiQA Architecture

Brini et al. (2009) proposed a QA system called QASAL that deals with factoid questions in Arabic using NLP. The QASAL system has an architecture containing three modules: First, the question analysis part receives the question from the user and focuses on the NER related to this question. This helps the system to know what type of answers to expect and facilitates answer extraction. For instance, when asked, "مـتـى اســـــتـقـلـت تـونـس؟" ("When did Tunisia become independent?"), the module knows the answer type is a time entity, which means to focus on the verb "استقلت" (independent) and the noun "تونس" (independent). Second, passage retrieval is the main part of this system, which returns the expected passages/documents that are related to the expected answer. If the system fails to return the expected right passages, this leads to low accuracy. Finally, the answer extraction module works to extract the answer from proposed passages that contain the answer. This module should execute the process differently according to the type of question. The authors used the Nooj platform to annotate the question with linguistic structure and regular expressions. The result showed 94% precision and 100% recall.

Al-Khawaldeh (2015) presented an approach to improve "why" questions' accuracy through developing the Entailment Adaptation in Arabic Why QA System (**EWAQ**). The idea was built based on re-ranking the relevant passages retrieved from other search engines, such as Google, Yahoo, and Ask. The system uses an entailment similarity algorithm between the retrieved documents/passages and the why questions. The architecture is built according to the three main components of questions analysis, answer extraction, and document retrieval. The system achieved good accuracy (68.53%) compared with other search engines.

The Arabic QA system called **AQuASys** presented by Bekhti and Alharbi (2013) accepts questions from users in natural language and retrieves the correct answer. Actually, the system depends on NER (location, person, organization, time, etc.) to return the right answer to a posed question. The system accepts the questions ("Who من","Where اين","When متى","What ما"), the questions are analysed to know the answer type and keywords, and then the relevant filtered sentences are retrieved from the documents. Moreover, the sentences are filtered based on the number of keywords in the question. The authors developed formulas to measure the similarity. The scoring phase performs two tasks: first, computing the scores of keywords in questions and sentences and the similarity of questions and sentences and occurrences of keywords in a question. Second, scoring function computing the final score according to the type of the asked question utilizing the rules. The below Figure 3.5 presents the system architecture and the main components. The results present 66.25% precision for 80 questions with excellent recall, reaching 97.5%.

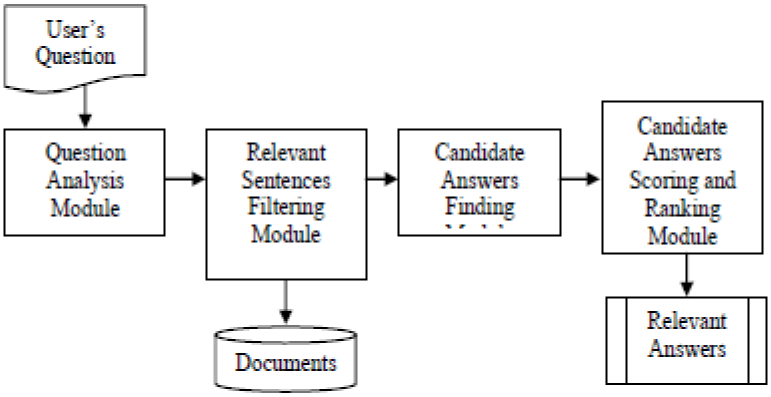

User's Question
Question Analysis Module
Relevant Sentences Filtering Module
Candidate Answers Finding Module
Candidate Answers Scoring and Ranking Module
Documents
Relevant Answers

Figure 3.5: AQuASys Architecture

3.4. QA Paradigms and Classification

The two main paradigms of question answering are IR-based factoid and knowledge-based. According to Jurafsky and Martin (2008), these paradigms retrieve the answers that are related to scientific reality. IR-based factoid has a gap between questions and answers. This creates a challenge in QA automation. Factoid questions spanned this gap by reformulating questions. In an effort to response the users' questions, IR-based QA aims at finding text parts in the passages list. The second paradigm of QA is of knowledge-based questions. According to Jurafsky and Martin (2008), semantic parsers use a logical form by mapping the question text to another format, for example, SPARQL.

Jurafsky and Martin (2008) stated that QA classification aims at identifying the answer type. For instance, for the question "Who is the queen of the UK?" the system establishes PERSON answer. Accordingly, the system can retrieve a close noun phrase inside the passage or text instead of analysing each sentence.

Mishra and Jain (2016) developed some criteria to categorize the questions in QASs. Firstly, **application domain** determines that the question is related to a certain application domain through terminology or ontology to return the suitable techniques to get the right answer in the same domain. According to Dhanjal and Sharma (2015), questions in open domains are not restricted and might have linkage with any concept. On the other hand, restricted-domain questions are directly linked with a specialist domain (disease, education, sports, etc.). Secondly, **types of questions** QAS depend on multiple techniques to return the right answer. Moldovan et al. (2003)

declared that mis-classification resulted in 36.4% of incorrect answers. This points to the impact of questions classification on the accuracy of an answer. Accordingly, Dhanjal and Sharma (2015) made a comparation in the below Table 3.1 between methods types in question classification.

<table>
<tr><th>Question Category</th><th>Pattern</th><th>Description</th><th>Example</th></tr>
<tr><td><b>Functional Word Questions</b></td><td>Begin with nonsignificant verb</td><td>All “Non-Wh” questions are considered under this category (except how).</td><td>Name the longest river in the world</td></tr>
<tr><td><b>How Question</b></td><td>Have two patterns
1)‘‘How + [does/do/did/AUX] + NP + VP + X?”
2)‘‘How [far|fast|long|much |many|big] + X?”</td><td>The answer type of the first pattern is the illustration of some process, whereas the second pattern retrieves figures as a result.</td><td>How does this work?
How long did you stay in that city?</td></tr>
<tr><td><b>Why Questions</b></td><td>Why + [does| do|did|AUX] + NP + [VP] + [NP] +” X”</td><td>The user asks for specific reasons or explanations.</td><td>Why is he tired?</td></tr>
<tr><td><b>Where Questions</b></td><td>Where + (do|does|did| AUX) + NP + VP + X?</td><td>It starts with the keyword “where” and represents natural locations.</td><td>Where is Paris located?</td></tr>
<tr><td><b>Who/Whom/Whose Questions:</b></td><td>(Who|Whom|Whose) + [does|do|did|AUX] + [VP] + [NP] + X?</td><td>These types of questions normally ask about an organization or an individual.</td><td>Who was that boy?
Whose bag is this?</td></tr>
<tr><td><b>Which Questions</b></td><td>Which + NP + X?</td><td>The answer type depends on the entity type of the NP.</td><td>Which company is in the top 20 market?</td></tr>
<tr><td><b>When Questions</b></td><td>When + (does|do|did|AUX) + NP + VP + X</td><td>It starts with the keyword “when” and is temporal in nature.</td><td>When did you finish your work?</td></tr>
<tr><td><b>List Questions</b></td><td></td><td>The expected answer is a list.</td><td>Name the famous actors in your country.</td></tr>
</table>

Table 3.1: Comparison between questions categories

Mishra and Jain (2016) divided questions into five types: (1) Factoid questions, which depend on facts and require a short sentence. Dang, Kelly and Lin (2007) said this question normally beings with “Wh” (What, Who, When, Which). (2) List questions: These questions are looking for multiple-fact answers. This requires a system to gather the answers from inside documents (Dang, Kelly, & Lin 2007). (3) Hypothetical questions: These questions demand information established with a hypothetical event; the IR technique is required to retrieve the answer (Mishra & Jain 2016). (4) Causal questions: This question generally begins with “how” or “why”. The NLP technique is required in this case for language analysis. The last type is (5) confirmation questions, which require one answer only (Yes/No).

The third type of criterion is **analysis type**, which includes three methods: rule-based, hybrid approach, and statistical-based. Mishra and Jain (2016) considered five types for this analysis (morphological analysis, syntactic analysis, semantic analysis, discourse and pragmatic analysis, and predictable answer). Manning et al. (2014) said that morphological analysis is an element of computational linguistics to divide words into morphemes and allot a class to a morpheme. Damljanovic, Agatonovic and Cunningham (2010) clarified syntactic analysis as the syntactic patterns that are frequently gained from an enormous dataset to work proficiently. Semantic analysis excerpts the possible meaning of questions according to questions’ words; the question parse tree interprets the meaning of the question. Quaresma and Rodrigues (2005) explained discourse and pragmatic analysis by using the pragmatic interpretation as a logical inference or ontology to produce the concepts and knowledge base from text. Predictable answer defines the named entity which is expected to be in the answer based on questions type (Mishra & Jain 2016).

The fourth type of criterion is **data type**, which designates the data sources as unstructured or structured. Mishra and Jain (2016) categorised data sources in QASs as unstructured, structured, and semi-structured. An unstructured data source means the data are presented in the dataset without any semantic style or storage rules, while a structured data source stores semantic style and uses entities and attributes to retrieve the answer. Semi-structured data sources rely on schemata to reach the data source.

The fifth criterion is data source properties, which analyse the text in multiple ways according to some properties, for example, source size, type, and language. Finally, techniques provide an indicator of questions’ complexity and the proper techniques for retrieving answers.

3.5. Summary

This literature review chapter illustrated the main background for QA. We discussed the meaning of NLP. In addition, Arabic challenges were explained in detail, and Arabic word analysis was illustrated with supported examples. Furthermore, QASs and Arabic QASs were explained, and the systems currently used were listed.

The literature shows a lot of Arabic challenges in QASs, which motivates us to answer research question 1. Also, the reviewed studies' results did not reach acceptable performance. Therefore, we will answer research questions 1, 2, and 3 to achieve acceptable performance for a QAS and enrich the rules-based method in QA methodology.

4. Chapter Four: Automatic Construction of Ontology from Webpages

4.1. Overview

As discussed earlier in Chapter Two, SW is important to building an ontology for complicated webpages. Generally, Arabic ontology research on Arabic e-government services is still limited and has not fulfilled the expected experimental results. In addition, the current ontologies are constructed for specific domains. This indicates the need to construct an ontology for the specific domains that are critical to evaluating QA tasks. Additionally, the main challenge of building an ontology for any domain is to conform to a standard, usually provided by a tool such as Protégé. This tool might lack flexibility in handling ontological components, which impacts quality.

In this chapter, we present the work to build a dependent ontology for Dubai government services and suggest a model for automatic construction of the ontology that depends on government entities' services' profiles. Moreover, it illustrates an approach to building an ontology in Protégé that follows the ontology development standard and tries to resolve its challenges.

4.2. Methodology

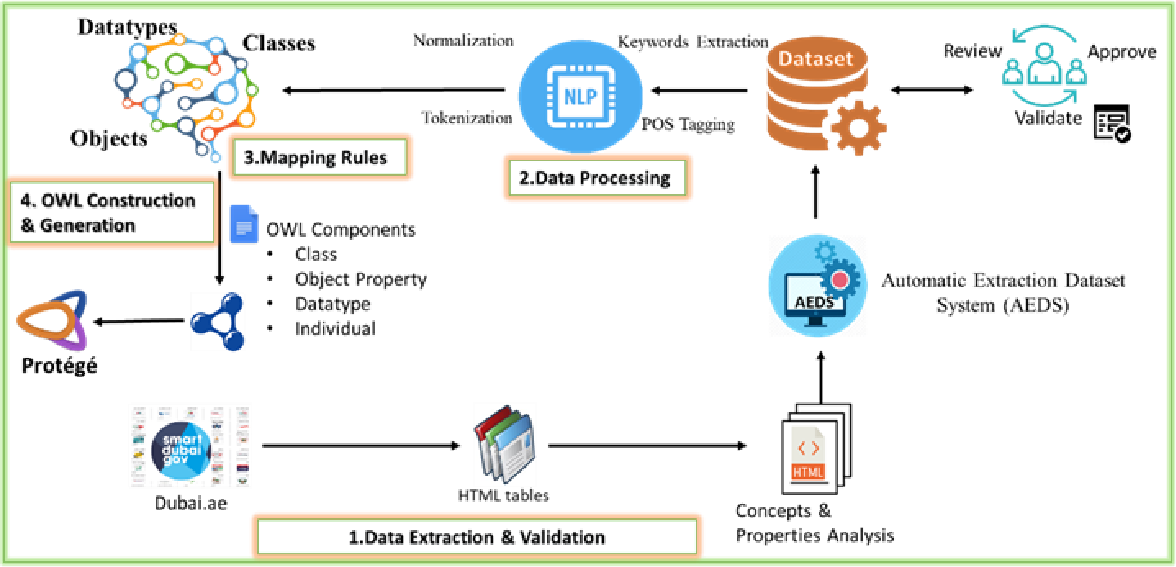


Figure 4.1: Ontology Construction Framework

Our ultimate objective is to automatically create an Arabic ontology from webpages and achieve significant results. Thus, we propose a methodology that relies on extracting the main entities, attributes, and relationships from webpages. In addition, we show how to automatically build an ontology through mapping between the extracted dataset structure and ontological components (classes, data properties, and data objects). In the proposed methodology, we develop a strategy to determine the ontological components such as the class structure, class attributes, and relationships from a set of webpages.

The proposed methodology is shown in Figure 4.1. The methodology has been designed to produce an automatic ontology for Dubai government services from their portal websites. This methodology is efficient in terms of time and effort. The proposed methodology consists of four main stages to build the ontology: data extraction and validation, data processing, mapping rules, and OWL construction and generation.

4.3. Data Extraction and Validation

This stage has a set of steps starting from data collection, document analysis, data extraction, and data validation. The input for this stage is the Dubai government portal, and the output is the dataset extracted from these webpages.

4.3.1. Data Collection

In this step, the data collection process is required to extract the ontology. Therefore, the scope of this research focuses on the Dubai government's services. Each government entity has its own services designed and implemented following the Dubai Model for Government Services (Dubai Government 2018). Each service usually includes some attributes in a formal channel: service name, service description, service channels, service delivery time, service procedures, services documents, service access, and contact channel (The Executive Council of Dubai 2015). The service directory consists of more than 26 pages and includes 531 services from 34 government entities.

Table 4.1 shows the list of government entities along with number of services.

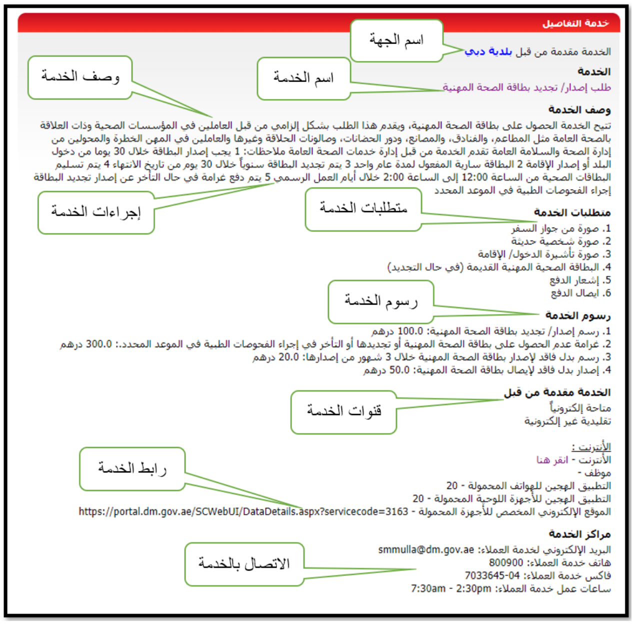
#	Government Entity	Number of Services
1	Dubai Civil Defence	12
2	General Directorate of Residency and Foreigners Affairs – Dubai	11
3	Dubai Airport Free Zone Authority	11
4	Public Prosecution – Dubai	20
5	Commission for Academic Accreditation	1
6	Emirates National Development Program	1
7	Dubai Municipality	30
8	Dubai Customs	37
9	Dubai Fishermen Cooperative Society	3
10	Land Department	23
11	Department of Economic Development	25
12	Department of Tourism and Commerce Marketing	10
13	Islamic Affairs and Charitable Activities	11
14	Dubai Cares	1
15	Dubai Police	18
16	Zakat Fund	2
17	Dubai Chamber	20
18	Dubai Courts	122
19	Dubai International Academic City	2
20	Dubai Healthcare City	2
21	Dubai Airports	6
22	Awqaf and Minors Affairs Foundation	4
23	Mohammed Bin Rashid Housing Establishment	9
24	Dubai Health Authority	50
25	Roads and Transport Authority	56
26	Knowledge and Human Development Authority	4
27	Community Development Authority	1
28	Dubai Culture	4
29	Dubai Electricity and Water Authority	25
30	Ministry of Higher Education and Scientific Research	5
31	Ministry of Social Affairs	1
32	Ministry of Labour	2
33	Dubai Government Human Resources Department	1
34	Dubai International Financial Centre	1
	Total	**531**

Table 4.1: List of Government Entities and Number of Services

4.3.2. Documents Analysis

After data collection, the documents require analysis to define the structure of the proposed ontology. The information inside each webpage is massive and complex. Thus, it requires proper understanding of its structure from information stored in a number of files that are in either Excel or XML format. The document analysis step has several tasks: (1) Ensure each service profile has a valid webpage, (2) identify the service attributes, (3) describe the service attributes, (4) identify the attribute IDs on webpages, and (5) map the service attribute with the identified ID.

Figure 4.2 explains the service webpage sample, giving an approach to understanding the service profile structure and extracting the main classes and information required of services that comply with the Dubai Model for Government Services. Each service profile page has nine attributes, as shown in Table 4.2, and each of them has its own information detailed on the webpage.


خدمة التفاصيل

الخدمة مقدمة من قبل بلدية دبي

الخدمة

طلب إصدار/ تجديد بطاقة الصحة المهنية

وصف الخدمة

تتيح الخدمة الحصول على بطاقة الصحة المهنية، ويقدم هذا الطلب بشكل إلزامي من قبل العاملين في المؤسسات الصحية وذات العلاقة بالصحة العامة مثل المطاعم، والفنادق، والمصانع، ودور الحضانات، وصالونات الحلاقة وغيرها والعاملين في المهن الخطرة والمحولين من إدارة الصحة والسلامة العامة تقدم الخدمة من قبل إدارة خدمات الصحة العامة ملاحظات: 1 يجب إصدار البطاقة خلال 30 يوما من دخول البلد أو إصدار الإقامة 2 البطاقة سارية المفعول لمدة عام واحد 3 يتم تجديد البطاقة سنوياً خلال 30 يوم من تاريخ الانتهاء 4 يتم تسليم البطاقات الصحية من الساعة 12:00 إلى الساعة 2:00 خلال أيام العمل الرسمي 5 يتم دفع غرامة في حال التأخر عن إصدار تجديد البطاقة إجراء الفحوصات الطبية في الموعد المحدد

متطلبات الخدمة

1. صورة من جواز السفر
2. صورة شخصية حديثة
3. صورة تأشيرة الدخول/ الإقامة
4. البطاقة الصحية المهنية القديمة (في حال التجديد)
5. إشعار الدفع
6. ايصال الدفع

رسوم الخدمة

1. رسم إصدار/ تجديد بطاقة الصحة المهنية: 100.0 درهم
2. غرامة عدم الحصول على بطاقة الصحة المهنية أو تجديدها أو التأخر في إجراء الفحوصات الطبية في الموعد المحدد.: 300.0 درهم
3. رسم بدل فاقد لإصدار بطاقة الصحة المهنية خلال 3 شهور من إصدارها: 20.0 درهم
4. إصدار بدل فاقد لإيصال بطاقة الصحة المهنية: 50.0 درهم

الخدمة مقدمة من قبل

متاحة إلكترونياً
تقليدية غير إلكترونية

الأنترنت :
الأنترنت - انقر هنا
موظف -
التطبيق الهجين للهواتف المحمولة - 20
التطبيق الهجين للأجهزة اللوحية المحمولة - 20
الموقع الإلكتروني المخصص للأجهزة المحمولة - https://portal.dm.gov.ae/SCWebUI/DataDetails.aspx?servicecode=3163

مراكز الخدمة

البريد الإلكتروني لخدمة العملاء: smmulla@dm.gov.ae
هاتف خدمة العملاء: 800900
فاكس خدمة العملاء: 7033645-04
ساعات عمل خدمة العملاء: 7:30am - 2:30pm

Figure 4.2: Service Profile Example

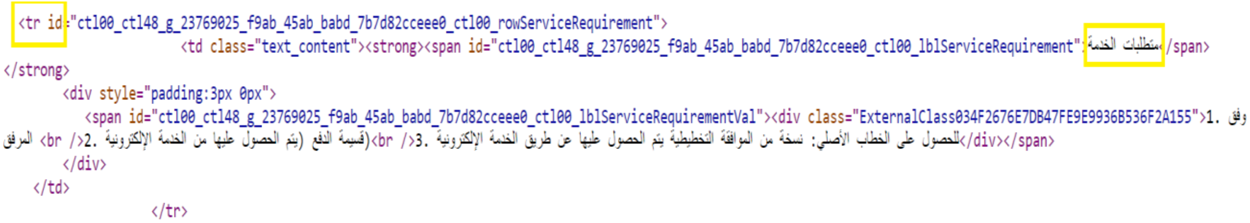
#	Service Attribute	Arabic Attribute Name
1	Entity Name	اسم الجهة
2	Service Name	اسم الخدمة
3	Service Description	وصف الخدمة
4	Service Fees	رسوم الخدمة
5	Service Channel	قنوات الخدمة
6	Service URL	رابط الخدمة
7	Service Document	متطلبات الخدمة
8	Service Procedure	إجراءات الخدمة
9	Service Contact	الاتصال بالخدمة

Table 4.2: Service Attributes

After analysing the webpages, we found that 94% of the collected services contain web links, i.e., URLs. All URLs are stored in a text file to locate the address of each service.

Furthermore, an analysis was performed for the Hypertext Markup Language (HTML) files for a sample of services to define the main concepts and attributes and map them to the ontology components. For example, each service attribute has a unique identifier (ID) located in in the HTML file as a table row.

Figure 4.3 shows an example of the HTML source code for the service “Request for planning permits”, which has the “Service Document متطلبات الخدمة” attribute. While analysing the HTML source code file, all data required in the HTML file are specified correctly and listed in a text file to be used in the data extraction phase in order to read each “ID” in the table row “<tr>” once and extract the information for each attribute.



```
 <tr id="ctl00_ctl48_g_23769025_f9ab_45ab_babd_7b7d82cceee0_ctl00_rowServiceRequirement">
                      <td class="text_content"><strong><span id="ctl00_ctl48_g_23769025_f9ab_45ab_babd_7b7d82cceee0_ctl00_lblServiceRequirement">متطلبات الخدمة</span>
</strong>
        <div style="padding:3px 0px">
          <span id="ctl00_ctl48_g_23769025_f9ab_45ab_babd_7b7d82cceee0_ctl00_lblServiceRequirementVal"><div class="ExternalClass034F2676E7DB47FE9E9936B536F2A155">1. وفق
المرفق <br />2. قسيمة الدفع (يتم الحصول عليها من الخدمة الإلكترونية)<br />3. للحصول على الخطاب الأصلي: نسخة من الموافقة التخطيطية يتم الحصول عليها عن طريق الخدمة الإلكترونية</div></span>
        </div>
    </td>
                </tr>
```

Figure 4.3: Sample Part of HTML Source Code File for Service

Table 4.3 is an example of one service’s attributes and IDs extracted from its HTML file.

ID	Attribute	Description
Label1	اسم الجهة	This attribute aims to identify the name of the entity and link it with the service. The defied rules are: • Mandatory field • Each service linked with the entity • Each entity has at least 1 service • Entity is related to Dubai government • Entity exists within Dubai.ae service webpages
lblServiceName	اسم الخدمة	This attribute is related to the service name; it has a set of attribute chains that should be linked to it. The defined rules are: • Mandatory field • Each service has at least 1 attribute • Each service should be linked to the entity • Each service has its own hyperlink
lblServiceDesc	وصف الخدمة	This is an attribute of a service to describe the service well. The defined rules are: • Optional field, linked to service chain (SC)
rowServiceRequirement	متطلبات الخدمة	This is an attribute of the service to present all requirements to deliver the service (for example, this service requires the following documents or information: invoice, passport copy, ID, etc.). The defined rules are: • Optional field, linked to service chain (SC)
rowServiceFees	رسوم الخدمة	This is an attribute of the service to inform the customer regarding the fees to deliver the service. The defined rules are: • Optional field, linked to service chain (SC)
rowServiceProcedure	إجراءات الخدمة	This is an attribute of the service to list the steps or process required to complete the service request. The defined rules are: • Optional field, linked to service chain (SC)
lblServiceChannel	قنوات الخدمة	This is an attribute of the service to inform the customer regarding the service delivery channels (online, centre, smart app, etc.). The defined rules are: • Optional field, linked to service chain (SC)
lblServiceCenter	مراكز الخدمة	This is an attribute of the service to inform the customer regarding the contact information or location of service customer centres. The defined rules are: • Optional field, linked to service chain (SC)
lblServiceUrl	رابط الخدمة	This is an attribute of the service to inform the customer regarding the service’s web link. • Optional field, linked to service chain (SC)

Table 4.3: List of IDs of Service in HTML file

4.3.3. Data Extraction

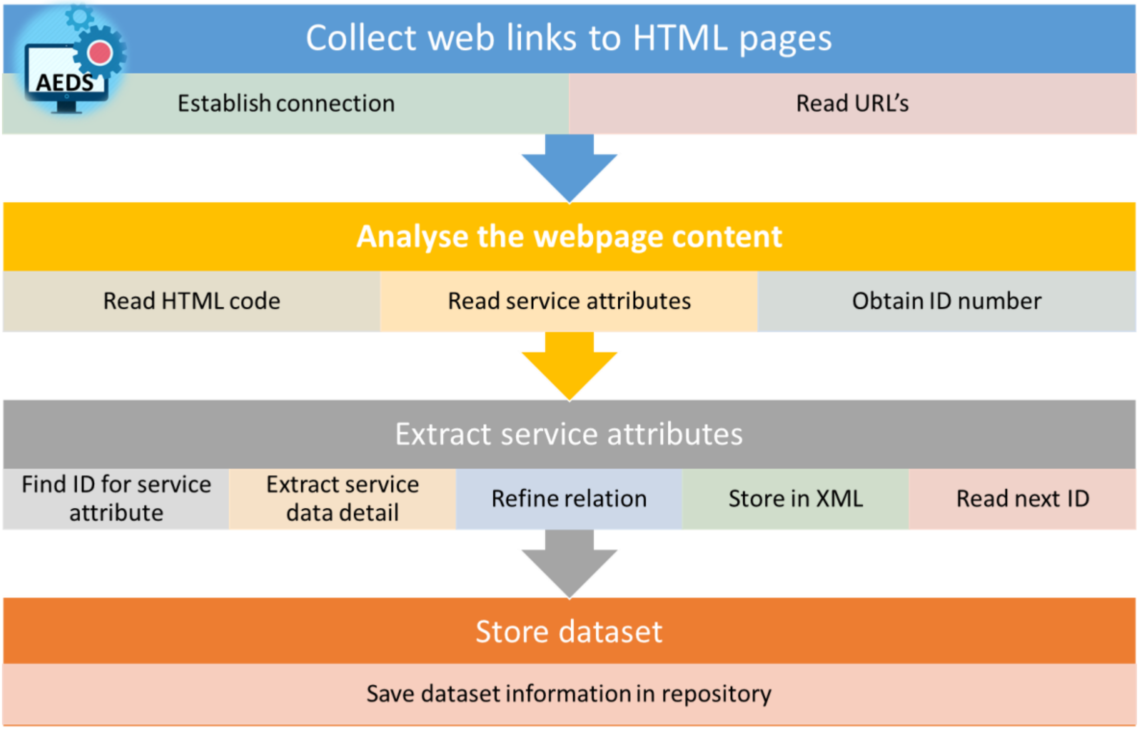

AEDS
Collect web links to HTML pages
Establish connection
Read URL's
Analyse the webpage content
Read HTML code
Read service attributes
Obtain ID number
Extract service attributes
Find ID for service attribute
Extract service data detail
Refine relation
Store in XML
Read next ID
Store dataset
Save dataset information in repository

Figure 4.4: Automatic Extraction Dataset System (AEDS)

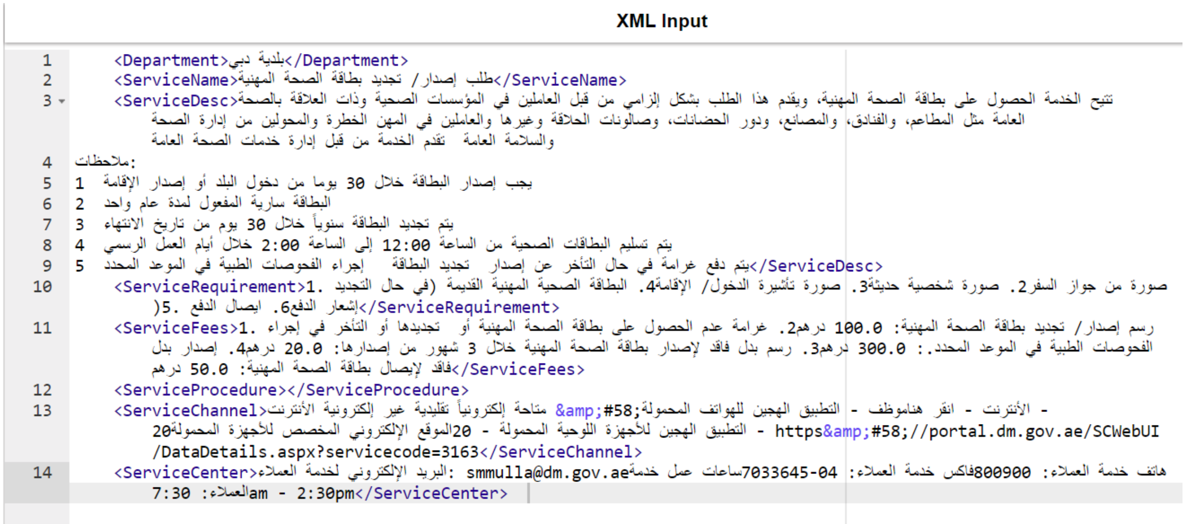
After collecting and analysing all target webpages that contain all services, we apply the Automatic Extraction Dataset System (AEDS), a tool developed to automatically extract all services' information from the collected webpages. Figure 4.4 illustrates the AEDS tool as described in the stages below:

1) **Collect web links to HTML pages**: This stage is to collect HTML pages by opening a connection with an HTML browser and reading the hyperlinks from the portal website.
2) **Analyse the webpage content**: Each webpage has HTML code, which requires analysis to define the main individual objects for each service. In HTML, the ID attribute specifies a unique ID for each service attribute within the HTML document, which acts like a primary key in the databases for each service. We used this information for relationship mapping across all service attributes.
3) **Extract service attributes**: In this step, we build the service record that contains the attributes according to service attribute ID for each fetched URL. The service chain (SC) equation is used to define the attribute for each service:

$$SC(p_i, p_j, u=1,2,3,\ldots,n).$$

The parameter p_i is used for service attribute, and p_j is used for service value, while u is used for the service's URL.

4) **Store dataset**: Each SC will be stored in XML/Excel format as a dataset for services' attributes, as shown in Figure 4.5.



XML Input

```
    <Department>بلدية دبي</Department>
    <ServiceName>طلب إصدار / تجديد بطاقة الصحة المهنية</ServiceName>
    <ServiceDesc>تتيح الخدمة الحصول على بطاقة الصحة المهنية، ويقدم هذا الطلب بشكل إلزامي من قبل العاملين في المؤسسات الصحية وذات العلاقة بالصحة
        العامة مثل المطاعم، والفنادق، والمصانع، ودور الحضانات، وصالونات الحلاقة وغيرها والعاملين في المهن الخطرة والمحولين من إدارة الصحة
        والسلامة العامة  تقدم الخدمة من قبل إدارة خدمات الصحة العامة
ملاحظات:
1  يجب إصدار البطاقة خلال 30 يوما من دخول البلد أو إصدار الإقامة
2  البطاقة سارية المفعول لمدة عام واحد
3  يتم تجديد البطاقة سنوياً خلال 30 يوم من تاريخ الانتهاء
4  يتم تسليم البطاقات الصحية من الساعة 12:00 إلى الساعة 2:00 خلال أيام العمل الرسمي
5  يتم دفع غرامة في حال التأخر عن إصدار  تجديد البطاقة  إجراء الفحوصات الطبية في الموعد المحدد</ServiceDesc>
    <ServiceRequirement>1. صورة من جواز السفر2. صورة شخصية حديثة3. صورة تأشيرة الدخول/ الإقامة4. البطاقة الصحية المهنية القديمة (في حال التجديد
        )5. ايصال الدفع. 6إشعار الدفع</ServiceRequirement>
    <ServiceFees>1. رسم إصدار/ تجديد بطاقة الصحة المهنية: 100.0 درهم2. غرامة عدم الحصول على بطاقة الصحة المهنية أو  تجديدها أو التأخر في إجراء
        الفحوصات الطبية في الموعد المحدد.: 300.0 درهم3. رسم بدل فاقد لإصدار بطاقة الصحة المهنية خلال 3 شهور من إصدارها: 20.0 درهم4. إصدار بدل
        فاقد لإيصال بطاقة الصحة المهنية: 50.0 درهم</ServiceFees>
    <ServiceProcedure></ServiceProcedure>
    <ServiceChannel>الأنترنت الإلكترونية غير تقليدية إلكترونياً متاحة &amp;#58;المحمولة للهواتف الهجين التطبيق - انقر هاموظف - الأنترنت -
        20المحمولة للأجهزة المخصص الإلكتروني الموقع20 - المحمولة اللوحية للأجهزة الهجين التطبيق - https&amp;#58;//portal.dm.gov.ae/SCWebUI
        /DataDetails.aspx?servicecode=3163</ServiceChannel>
    <ServiceCenter>هاتف خدمة العملاء: 800900فاكس خدمة العملاء: 04-7033645ساعات عمل خدمة العملاء: 7:30am - 2:30pm البريد الإلكتروني لخدمة العملاء: smmulla@dm.gov.ae</ServiceCenter>
```

Figure 4.5: Service Chain Sample

According to the extracted dataset from the AEDS system, statistical analysis has been performed to evaluate the extracted webpages' contents and validate them in the next phase. The bar chart in Figure 4.6 gives statistical information about service attributes.

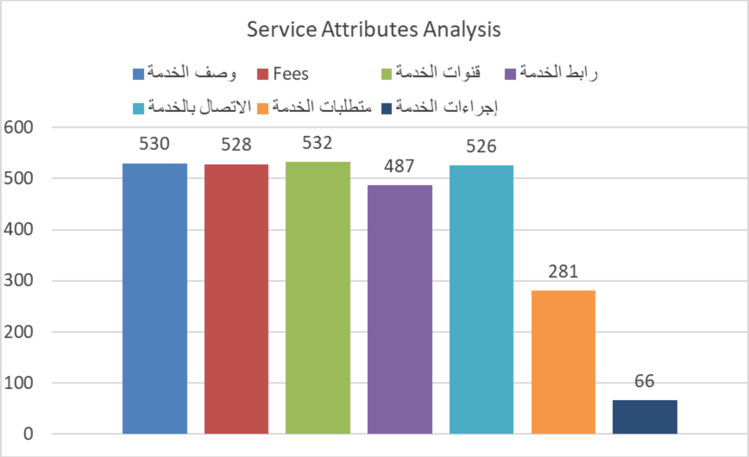


Figure 4.6: Number of Extracted Attributes from All Services

4.3.4. Data Validation

After extracting the dataset, we have to validate its quality before we start processing and building the ontology. Therefore, a team of two domain experts from a government entity in Dubai has conducted a manual validation process. The first expert has experience in service standards according to the Dubai Model, and the second expert is a service analyst from a government entity. Additionally, to help the domain experts, a user manual has been provided to them so they can understand the validation process. The main processes for validation as shown in Figure 4.7 are review dataset, validate services, validate attributes, and approve/reject service record by tagging.

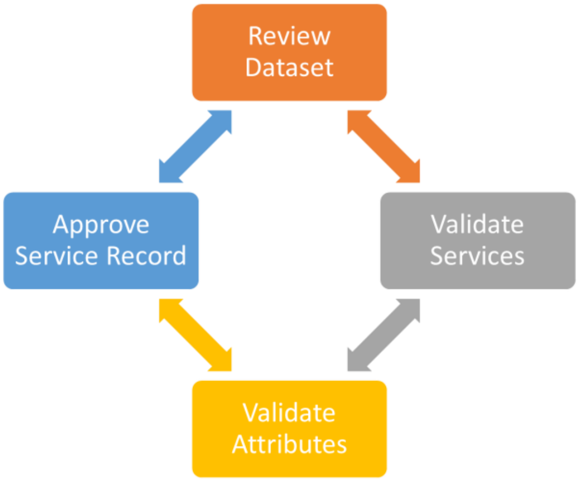


Figure 4.7: Validation Dataset Process

Each process has multiple sets of activities.

1) **Review Dataset:**
    a. Review dataset structure.
    b. Ensure all attributes are captured.
    c. Count the number of services for each entity.
2) **Validate Services:**
    a. Ensure the service is related to the entity.
    b. Ensure each service has the correct attributes.
    c. Measure the quality of service information.
    d. Mark the number of issues for each service.
3) **Validate Attributes:**
    a. Check that the attributes are related to the service.

b. Validate the attributes' contents (documents, fees, etc.).

c. Mark the number of issues for each attribute.

4) **Approve Service Record:**

a. Check 1 if all information is correct and valid.

b. Check 0 if there is some missing or invalid information.

4.3.5. Inter-Annotator Agreement

To resolve conflicts in the validation process, inter-annotator agreement is used between the domain experts, and it gives us an indication of the quality of the extracted data before we start to build an automatic ontology. Alqaryouti, Siyam and Shaalan (2019) stated that inter-annotator agreement is essential to ensure the transparency of annotators and verify their proper understanding, consistency, and quality. The best-known method for inter-annotator agreement is kappa (McHugh 2012), which tests interrater reliability.

McHugh (2012) stated that Cohen's kappa statistics are used for two annotators; the results of the interrater test range between -1 and +1, where the value 0 means null agreement, and 1 means perfect agreement. Equation *1* shows the formula to calculate the results. $\mathcal{K}$ is kappa, $\mathcal{P}\mathrm{r}(\mathrm{a})$ is an observed agreement, and $\mathcal{P}\mathrm{r}(e)$ is a chance agreement.

$$k = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)}$$

Equation 1: Kappa Formula

In addition, the author prepared a statistical measurement table as below Table 4.4.

Value	**Agreement Results**
<= 0	No Agreement
0.01-0.20	Slight Agreement
0.21-.40	Fair Agreement
0.41-0.60	Moderate Agreement
0.61-0.80	Substantial Agreement
0.81-1.00	Perfect Agreement

Table 4.4: Cohen's Kappa Statistics Measurement

These measures produce a Cohen’s kappa ($\mathcal{K}$) of 89.08%, which indicates that our results are perfect based on Takala et al.’s (2014) annotators’ matrix. Therefore, the results have been approved and meet confidence requirements with high quality according to the baseline information.

4.4. Data Processing

In this stage, the extracted dataset represents a knowledge base for services and should have an information process to build an ontology with high-quality information. All concepts in the dataset are subject to the following NLP tasks: normalization, part-of-speech tagging, stop word removal, and keyword extraction.

4.4.1. Normalization

Normalization is one of the challenges in Arabic. The problem appears because of the inconsistencies in writing Arabic script, such as the hamza letter, diacritic marks such as the madda symbol, and specific symbols such as the dot (Farghaly & Shaalan 2009). The main purpose of normalization is to (1) decrease the processing time by removing the Arabic special characters and (2) replace the space character with "_" for service and entity classes attributable to the Protégé system’s limitations in handling Arabic. The characters that need to be removed from service and entity are “،”,”,”,”.”,”/”,” \”,”!”,” ؟”, etc., mostly punctuation marks. However, the URL attribute is an exception because it is stored in English. As a result, of the dataset’s 70,042 words, 19.06% were normalized.

For example, the normalization process for the service " طلب إصدار/ تجديد بطاقة الصحة المهنية" removes the character “/” and replaces the spaces with “_”. The output of the normalization process is "طلب_إصدار_تجديد_بطاقة_الصحة_المهنية".

4.4.2. Part-of-Speech (POS) Tagging

Basically, it is important to use NLP features to extract information from text. Therefore, we use the part-of-speech tagging technique (Chang et al. 2006). A POS tagger is used to identify the type of each word in the service name class and the entity name. The output of this process is the category of each word (verb, adjective, noun, preposition, etc.; Darwish & Mubarak 2016). The dataset was uploaded online on the FARASA[4] tool (Abdelali et al. 2016). The output of the extracted dataset for the service class before normalization shows 2,711 words: 324 adjectives, 1812 nouns, and 263 verbs, and the rest are prepositions, pronouns, and punctuation. As per analysis of the POS results, there were 151 words tagged as nouns and corrected to verbs from customer perspective, including “Issue” “إصـــــدار”, “Export” “تصـــــدير”,” Pay” “دفع”,” License” “تــرخــيــص”, etc. In addition, there were eliminated and discarded prepositions, conjunctions, punctuation, and pronouns. The outputs of this process for service class before normalization are presented in Table 4.5.

POS Tag	Number of words	Example
ADJ	324	بسيطة، الأخرى، الهندسية
CONJ	25	و، أو، ف
FOREIGN	5	نْe , \
NOUN	1812	شهادة، الأجهزة، الكهرباء
NUM	1	,
PART	6	على، التي، الذي
PREP	179	عن، في، من
PREP+PRON	2	بها
PRON	1	ذلك
PRON+PRON+PRON	3	يهمه
PUNC	90	/،)، -
V	263	طلب، عرض، تتبع
Grand Total	**2711**	

Table 4.5: POS Tagging for Service Class

4.4.3. Keywords/Terms Extraction

Keyword extraction tokenizes the text into keywords, tags each keyword by POS, then stems the keywords to their roots (Ray & Shaalan 2016). For instance, the service text "طلب_تحويل_شحنات_غذائية_من_وإلى_الإمارات_الأخرى" has four keywords: "طلب","تحويل" ,"شحنات" and "الإمارات". Ordinarily, the ontology structure is designed based on the classes, data properties, and data objects. Therefore, our methodology in ontology construction is built according to the keywords/terms, where each service is linked to the related keywords, as mentioned in the previous example.

[4] http://qatsdemo.cloudapp.net/farasa/

A statistical parsing approach is adopted to extract the key phrases/keywords according to Dostal and Ježek (2011). After the keywords are extracted, new rules are built between keywords and entities/services to empower the relations between them. Table 4.6 shows the relations of the service "استعراض_و_دفع_مخالفات_سالك" with the entity and keywords.

Entity	Service	Keywords
هيئة_الطرق_والمواصلات	استعراض_و_دفع_مخالفات_سالك	استعراض
		مخالفات
		سالك
		دفع

Table 4.6: Entity/Service Keywords Relations

All extracted keywords are stored in XML format for mapping purposes. The final results of the extracted keywords in simple terms consist of 414 verbs, 1,739 nouns, and 339 adjectives. Figure 4.8 presents the word cloud of the top keywords used in the services class.

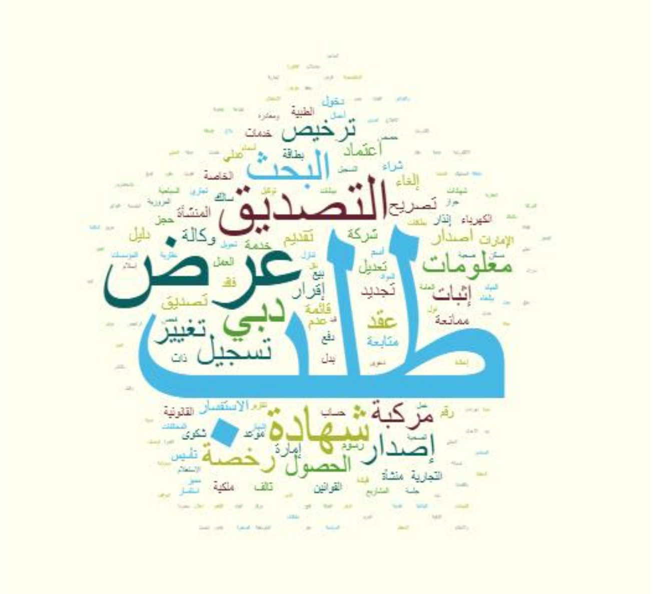


Figure 4.8: Word Cloud of the Most-Used Keywords in the Services

4.5. Mapping Rules

After the data processing stage, we start to transform the extracted dataset into the OWL ontology format (classes, object properties, and datatype properties). Therefore, the mapping stage is mandatory to define the base rules for Dataset-OWL mapping. We use same mapping classification rules used by Rodrigues, Rosa & Cardoso (2006):

- **Class mapping** maps an entity/service node to an OWL concept.
- **Datatype property mapping** maps service profile to an OWL datatype property.
- **Object property mapping** links the classes' rules to an OWL object property.

The ontology structure has three classes: Entity "الجهة الحكومية", Services "الخدمات", and Keywords "كلمات المفتاحية". Figure 4.9 explains the conceptual model of a service entity that helps in ontology mapping.

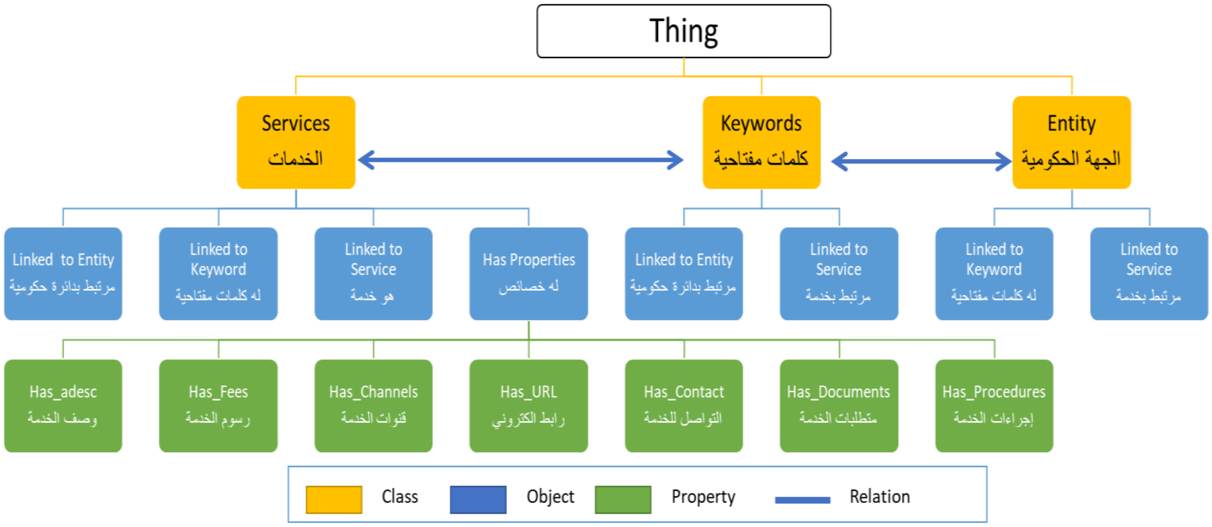


Figure 4.9: OWL Structure of Service Entity Ontology Mapping

Moreover, the data property for each class comprises the content information for each service. The service normally has seven properties, as shown in Table 4.7.

Property Name (EN)	**Property Name (AR)**	**Class**	**Data Type**
Has_adesc	وصف الخدمة	S_خدمة	String
Has_Fees	رسوم الخدمة	S_خدمة	String
Has_Channels	قنوات الخدمة	S_خدمة	String
Has_URL	رابط الخدمة الالكتروني	S_خدمة	String
Has_Contact	التواصل للخدمة	S_خدمة	String
Has_documents	متطلبات الخدمة	S_خدمة	String
Has_procedures	إجراءات الخدمة	S_خدمة	String

Table 4.7: Service Datatype List

This step is crucial for a solid ontology, which impacts the IR techniques used. Also, it increases the accuracy of QA systems, which depend on relations to return the right answers. The object property is a relation between ontology classes; this will be created based on defined classes and data properties. Each entity has set of services and keywords; hence, we have to define the relations to show these interactions. The main relations in this ontology are listed in Table 4.8:

Property/Relation	Connected to Class	Purpose
Is_Entity	Entity – الجهة_الحكومية	To check whether the object is an entity or not.
Linked_to_service	Service – الخدمة Entity – الجهة_الحكومية Keyword – كلمة مفتاحية	To link the service with the entity and related keywords.
Linked_to_Entity	Service – الخدمة Entity – الجهة_الحكومية Keyword – كلمة مفتاحية	To link the entity with related services and keywords.
Linked_to_Keyword	Service – الخدمة Entity – الجهة_الحكومية	To link each keyword with related services and entities.

Table 4.8: Ontology Relations

Figure 4.10 illustrates an example of relations between classes and objects. The service "Register Business تسجيل الأعمال" is defined as “service خدمة” class and connected to the entity called “Dubai Customs جمارك دبي”. Moreover, the service has two keywords: “Register تسجيل” and “Business الأعمال”. In addition, keywords and entity have the same relations in both directions, which makes the rules more efficient.

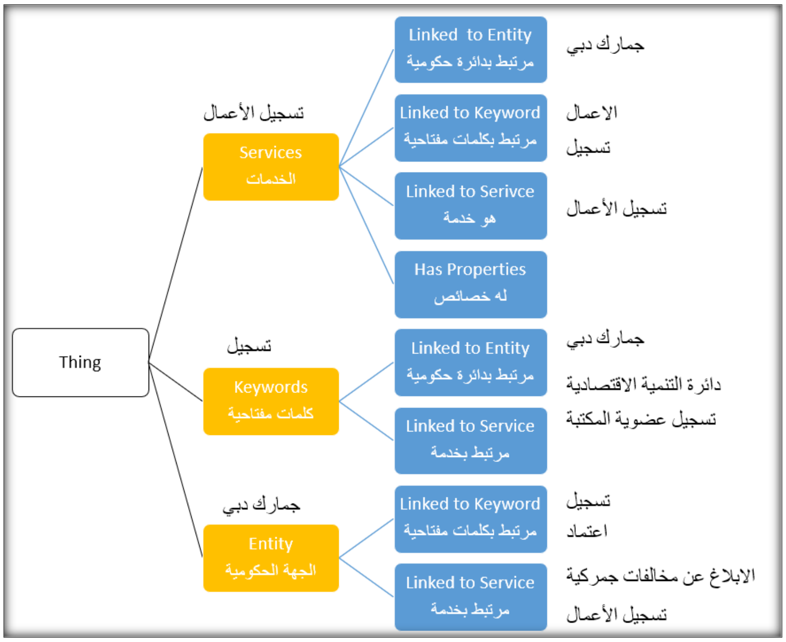
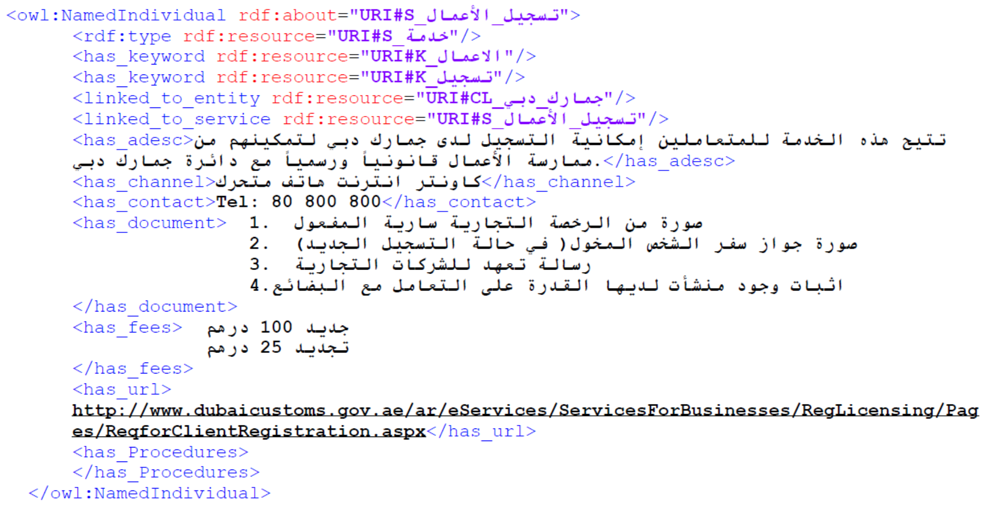


Figure 4.10: Objects Relations Example

In order to build the structure of the OWL ontology components, we need to define the structure of three parts: “NameIndividual”, “rdf:about”, and “rdf:resource”. The following example shows this process, and Figure 4.11 presents the output script for one class rules mapping:

1) Define the “NameIndividual” for the service name “تسجيل_الأعمال”. This class simply gives an alternative method to declare that a given entity is an individual.
2) Define the “rdf:about”, which points to the Uniform Resource Identifier (URI) of the ontology, which is used to identify the location of the identified ontology concepts and attributes. The URI used is “rdf:about="http://www.semanticweb.org/alibarg31/ontologies/2018/3/untitled-ontology-194#”.
3) Run the predefined mapping rules process to link the service name with the following OWL data objects: “Linked_to_Keywords”, “Linked_to_Entity”, and “Linked_to_Service”. The “rdf:resource” indicates the data element of data objects. Also, the same mapping rules process links the service attributes with OWL data properties.
4) Repeat the previous process for all ontology data objects. This process allows recording of all ontology components, storing them in an OWL format, and creating Resource Description Framework (RDF) triples.



```
<owl:NamedIndividual rdf:about="URI#S_تسجيل_الأعمال">
      <rdf:type rdf:resource="URI#S_خدمة"/>
      <has_keyword rdf:resource="URI#K_الاعمال"/>
      <has_keyword rdf:resource="URI#K_تسجيل"/>
      <linked_to_entity rdf:resource="URI#CL_جمارك_دبي"/>
      <linked_to_service rdf:resource="URI#S_تسجيل_الأعمال"/>
      <has_adesc>تتيح هذه الخدمة للمتعاملين إمكانية التسجيل لدى جمارك دبي لتمكينهم من
      ممارسة الأعمال قانونياً ورسمياً مع دائرة جمارك دبي.</has_adesc>
      <has_channel>كاونتر انترنت هاتف متحرك</has_channel>
      <has_contact>Tel: 80 800 800</has_contact>
      <has_document>  1.  صورة من الرخصة التجارية سارية المفعول
                      2.  صورة جواز سفر الشخص المخول( في حالة التسجيل الجديد)
                      3.  رسالة تعهد للشركات التجارية
                      4.اثبات وجود منشأة لديها القدرة على التعامل مع البضائع
      </has_document>
      <has_fees>  جديد 100 درهم
                  تجديد 25 درهم
      </has_fees>
      <has_url>
      http://www.dubaicustoms.gov.ae/ar/eServices/ServicesForBusinesses/RegLicensing/Pag
      es/ReqforClientRegistration.aspx</has_url>
      <has_Procedures>
      </has_Procedures>
  </owl:NamedIndividual>
```

Figure 4.11: An Example of OWL Service Script Rules Mapping

4.6. OWL Construction and Generation

There are several tools for building ontologies, such as DOML, Protégé, and Hozo. The most common and well-known tool is Protégé (Stanford Center 2016), which we decided to follow. To build an ontology, there are several main steps that should be followed to understand the structure of the OWL representation in the ontology: (1) Identify the ontology scope and domain, (2) identify the ontology's overall structure, (3) define the ontology classes, (4) define the ontology attributes/properties, (5) define the ontology data types, and (6) create the instances.

In the OWL format, there are three main class resources ("S_خدمة", "CL_الجهة_الحكومية", and "Keywords"). All the generated classes' scripts are collected in an OWL file. The property code has two types of properties that are required for ontology construction, namely, object and datatype. The object code is defined in an OWL file as "owl:ObjectProperty", and the datatype code, considered the value of the object, is defined as "owl:DatatypeProperty". Both of them are subclasses in the RDF class and are defined as "rdf:Property" (Yadav et al. 2016). The defined objects in OWL are "linked_to_keyword", "is_entity", "linked_to_entity", and "linked_to_service". The individual has a unique name: "Facts". After defining all classes and objects, the instances are created from the extracted dataset.

Figure 4.12 is an example of a generated individual in OWL form for the entity "CL_جمعية_دبي_التعاونية_لصيادي_الأسماك" as a graph. This entity has nine keywords and three services and is linked to "CL_الجهة_الحكومية". The datatype is a range of data values. OWL uses the

RDF data typing schema represented in XML format. As explained, each service class has seven properties. Our algorithm is applied to the extracted dataset and retrieves these properties, then links them to each service class.

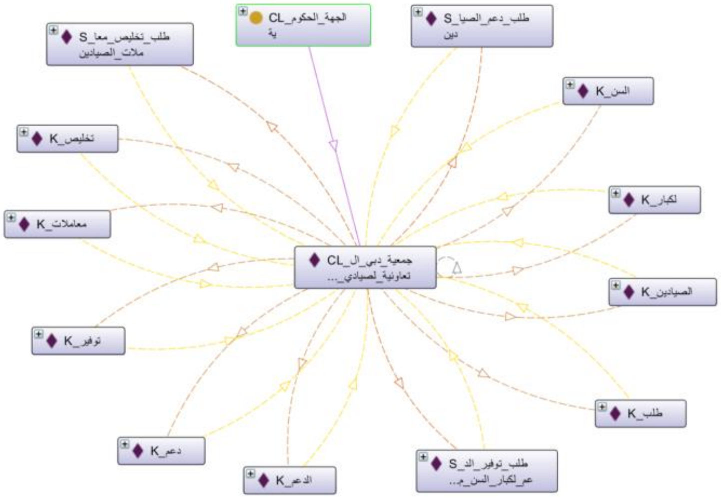


Figure 4.12: Ontology Graph nodes for Government Entity

4.6.1. Ontology Construction

The final step is the automatic ontology construction based on the OWL format. The generated classes, properties, individuals, datatypes, and header code make a collection that produces the ontology, which represents the knowledge in a graphical interface. Some errors occurred through opening the ontology in the Protégé tool. The main reasons were the representation structure and Arabic language challenges (for example, the space character, Unicode, and special characters). Further, the system rejected using more than one word as a class name or data object. This issue was resolved by replacing the space character with the underscore character "_".

Ontology statistics show the actual results of the ontology construction approach. Table 4.9 presents the ontology statistics.

File Size in KB	3,135 KB
Number of Lines	30,466
Number of Words	103,631
Number of Concepts/Triples	19,907
Number of Relations	13,011
Number of Domains	3
Number of Ranges	4
Number of Entities	34
Number of Properties	3,714
Number of “linked_to_keywords” Rules	4,024
Number of “linked_to_service” Rules	3,521
Number of “linked_to_entity” Rules	2,092
Number of “rdf:type” Objects	3,340

Table 4.9: Ontology Statistics

The last step is uploading the OWL file in the Protégé tool and drawing the ontology in visualization mode. Figure 4.13 presents the generated OWL of our ontology, representing the extracted knowledge in a graph with semantic relations between terms and concepts.

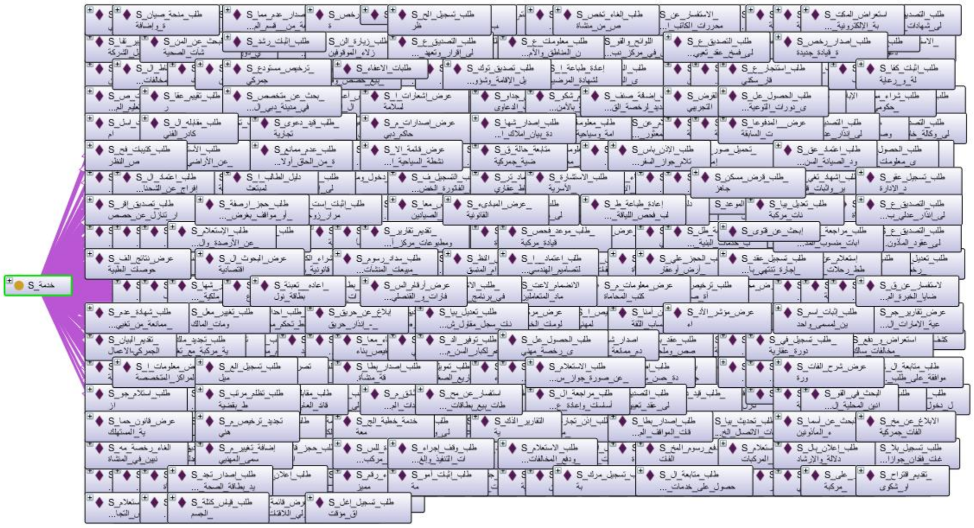


Figure 4.13: A graph representation of our ontology

4.7. Experimental Results and Discussion

In this section, the ontology evaluation has been conducted to ensure the quality and the accuracy of ontology elements (concepts, relations, and keywords). In the literature, these evaluation metrics have been used to evaluate similar research (Benabdallah, Abderrahim & Abderrahim 2017). Typically, the measurements used are precision, recall, and F1-measure, as displayed in Figure 4.14. The precision formula is the total number of right entities produced by our approach divided by the gross number of entities produced by our approach, while the recall formula is the total number of right entities produced by our approach divided by the gross number of right entities in the collected dataset. The F1-measure is defined as the harmonic mean of precision and recall. A confusion matrix is used to evaluate the concept/relations components in the extracted ontology.

$$Precision = \frac{\text{The total number of right entities produced by our approach}}{\text{The gross number of entities produced by our approach}}$$

$$Recall = \frac{\text{The total number of right entities produced by our approach}}{\text{The gross number of right entities in the collected dataset}}$$

$$F1 - measure = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Figure 4.14: Performance Measurement Formulas

Domain experts evaluated the extracted dataset before building the ontology, and inter-annotator agreement was applied to validate their evaluations. The same domain experts evaluated the ontology manually by tagging the correct concepts and relations levels. Table 4.10 shows the precision, recall, and F1-measure for ontology components (concepts and relations). The result shows 92% precision on average for concepts, and the recall achieves 97%, which indicates a high reliability level. The results of concept relations between individual objects and entities present on average 82% of extracted terms/keywords and entities/services, while the recall reaches 97%, which indicates a high level of linguistic techniques being used. The main reason for the low precision of the entities/services goes back to the accuracy of the rules between concepts. The overall average results are 87%, 97%, and 92% for precision, recall, and F1-measure, respectively.

Ontology Components	**Precision**	**Recall**	**F1-Measure**
Concepts	***92%***	***97%***	***94%***
Entities	91%	100%	95%
Services	93%	93%	93%
Relations	***82%***	***97%***	***88%***
Keywords	92%	94%	93%
Entities/Services	72%	100%	84%
Average	***87%***	***97%***	***92%***

Table 4.10: Experimental Results of Ontology Construction

We have come across various challenges throughout building and extracting the dataset and ontology extraction. For instance, in Arabic keyword extraction and segmentation, we used a statistical method to extract the key phrases/keywords, but this challenge requires a strong dataset for Arabic keywords to evaluate the extracted terms. In addition, there is a limited number of services webpages for Arabic resources, which impacts the quality of the data sources and the relations between the attributes of each service; this was resolved by developing our AEDS tool to extract the dataset. Finally, the Arabic POS tagger version that we used has some limitations: There were some words tagged as nouns despite actually being verbs. This issue was manually resolved with a post-processing step. However, there is still the potential to resolve it using other tools.

4.8. Summary

In this chapter, we proposed an approach to constructing an Arabic ontology for Dubai e-government customer services. In our approach, the entire Arabic services dataset was extracted from the dubai.ae portal using our developed AEDS tool. An analysis was performed on all collected webpages. Our methodology is divided into four stages, starting from data extraction and validation by using inter-annotator agreement to measure the quality of dataset, then using NLP tasks to extract the terms/keywords and cleaning the dataset. OWL/RDF representation is the base of any ontology using the Protégé tool. Finally, the experimental results showed significant performance, with high precision and recall for the extracted concepts and relations.

5. Chapter Five: Arabic Semantics-based Question Answering

Nowadays, huge amounts of information have become available in different structures because of variations in users, data, objectives, and systems. In addition, updating this information is extremely fast, which creates challenges in answering users' questions correctly. Researchers have competed to adopt the right technology to understand the meaning of a user questions in order to retrieve the right answer. Although a lot of research dealing with English and other languages has been published, Arabic research still suffers from some limitations due to the grammatical, semantic, and morphological complexity of the Arabic language. Therefore, in this chapter, we present Arabic QA using a semantics-based approach that depends on the e-government services ontology dataset extracted in Chapter Four. The purpose is to analyse users' questions using NLP techniques and translating them from NL to SPARQL queries in order to retrieve the correct answer via keyword-based and semantics-based approaches.

5.1. Methodology

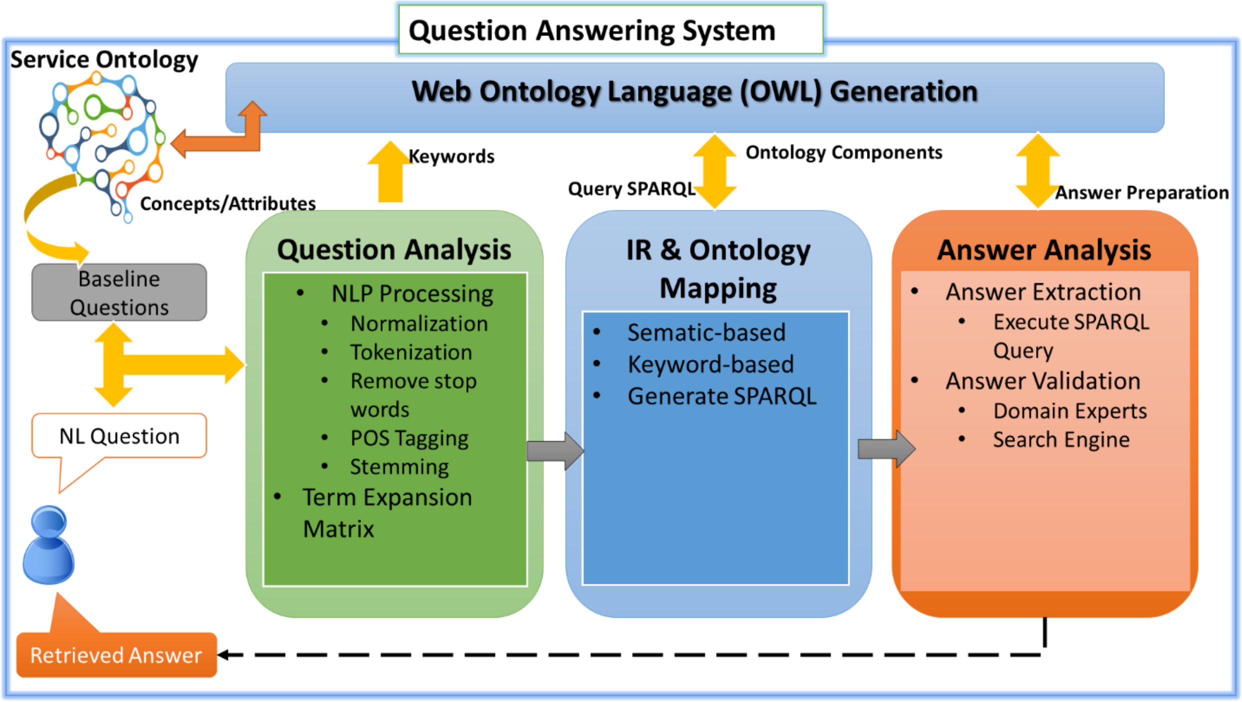


Figure 5.1: QA Methodology

The extracted ontology will represent the dataset for our QA methodology, as presented in Figure 5.1. This helps to ensure the quality of the ontology domain as well as to answer the NL queries relevant to e-government services. The QA architecture is a common framework recently published (Ray & Shaalan 2016). Figure 5.1 explains the QA methodology developed to answer user questions asked in NL that are related to Dubai’s government services. The methodology has four stages: preparing baseline questions, question analysis, IR and ontology mapping, and answer validation.

5.2. Baseline Questions

The main purpose of the QAS is to deliver a system to deal with computers in NL. Therefore, QAS does not require a programming background or any query language such as SPARQL or SQL to access the knowledge base. Nevertheless, the user must determine the ontology domain structure to ask questions. For that reason, we have prepared NL questions according to Tatu et al. (2016) that will be designed based on keywords/terms to retrieve the relevant answers. In addition, the domain experts’ questions will be using in the baseline questions. To identify the range of questions and answers, we derived questions from the objective that were applied to implement the ontology dataset, particularly the ontology domain and the main concepts that are used as the knowledge base of the ontology.

Basically, in Chapter Four, we constructed an ontology based on keywords. With the aim of measuring the ontology structure well, we decided to automatically build the questions based on ontology entities by using d’Aquin and Motta’s (2011) approach. The authors proposed a method to automatically build questions that might be answered by an ontology dataset. Formal concept analysis (FCA) was used to construct the questions based on the ontological entities and main keywords in multiple levels of user questions. There are 414 verbs in the ontology keywords’ concepts. These keywords were used to automatically build the unique questions. The template that was used to formulate a question is as follows:

(Question Type) (Attribute) (Ontology Class) (Individual) (in) (Entity)

To give an illustration of this formula, consider the keyword verb “ترخيص” of the service “ترخيص مستودع جمركي”, and that we need to ask about the attribute fees "رسوم" for this service. This attribute is a data property in this service that is linked with the entity “جمارك دبي”. Based on the proposed formula, the question is illustrated in Figure 5.2.

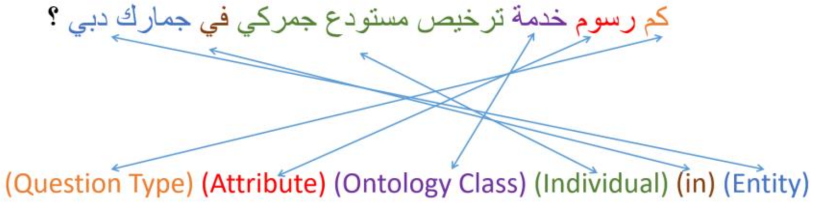

كم رسوم خدمة ترخيص مستودع جمركي في جمارك دبي ؟
(Question Type) (Attribute) (Ontology Class) (Individual) (in) (Entity)

Figure 5.2: FCA Example

Additionally, to ensure the performance of our approach, an online survey was performed with six domain experts in service management to ask them for five questions related to Dubai's government services. Hence, 30 questions were collected from this survey that are subject to the QA approach.

There are three types of standard questions considered when building questions: factoid, list and complex. Table 5.1 shows the baseline question categories based on the answer types, in addition to domain experts' questions.

Question Type	**Description**	**No. of Questions**	**Example**
Factoid	Question returns one answer.	309	ما هي ساعات العمل في هيئة الصحة بدبي؟
List	Question returns multiple answers.	64	اذكر خدمات دائرة السياحة والتسويق التجاري؟
Complex	Question has complex structure.	41	كيف يمكن الاستفسار عن خدمة طلب تعديل رسوم السكن في هيئة كهرباء و مياه دبي ومعرفة رسومها؟
Users	Domain experts' questions	30	كيف ادفع مخالفات سالك؟
Total	**Total number of questions**	**444**	

Table 5.1: Baseline Questions Types

In order to classify the questions properly, we use two types of question classification: taxonomy and semantic interpretation. Taxonomy relies on the form of the question. For instance, seven coarse types of English questions have been listed (where, who, what, why, which, when, and how) (Lahbari, El Alaoui & Zidani 2018). Table 5.2 presents the proposed English and Arabic question taxonomy based on the selected dataset.

Question type in English	Question type in Arabic
How	كيف، كيف يتم،
What	ما هي، ما
When	متى
Where	أين
List	اذكر

Table 5.2: Question Taxonomies

Since our focus on the government domain, we have to use an extra taxonomy type based on the semantic interpretation of the answer. For example, in "What are the fees of the service Register Client?" the coarse class is Number, and the defined "Fine Class" is the "Fee" attribute in the services class. In order to retrieve the correct attribute when the question is asked, we have prepared a matrix to link between the question type, the ontology class, and the required attribute. Table 5.3 illustrates the word/attribute term expansion matrix and the semantic interpretation based on "Fine Class".

Question type in English	Question type in Arabic	Main Class	Attribute
What are the documents…?	ما هي المستندات ...؟	S_خـدمـة, Cl_ الجهـة الحكومية	Has_document
What is the service…?	ما هي خدمة ...؟	S_خـدمـة, Cl_ الجهـة الحكومية	Has_adesc
What is the channel…?	ما هي قنوات ...؟	S_خـدمـة, Cl_ الجهـة الحكومية	has_channel
When does work start…?	متى يتم بدء العمل ...؟	S_خـدمـة, Cl_ الجهـة الحكومية	has_contact
How much are the service fees…?	كم رسوم خدمة...؟	S_خـدمـة, Cl_ الجهـة الحكومية	has_fees
List the service list…	اذكر خدمات...؟	S_خـدمـة, Cl_ الجهـة الحكومية	name
Where I can use…?	اين يمكن ان استخدم...؟	S_خـدمـة, Cl_ الجهـة الحكومية	has_url name

Table 5.3: Word/Attribute Term Expansion Matrix

In order to increase the accuracy and efficiency of answers, Arabic WordNet (Arabic WordNet 2018) is used to formulate Arabic queries. The below Table 5.4 shows the Arabic words most used in our dataset.

Term	Term Expansion
طلب	سأل، دعوة، التماس، عريضة
عرض	تقديم، اقتراح، إظهار، استعراض
تسجيل	دخول، قيد، انخراط
ترخيص	إذن، تصريح، تفويض
تصديق	قبول، اعتماد، توقيع
تجديد	إعادة، تأهيل، إصلاح، ترميم

Table 5.4: Top Terms in Service Class

5.3. Question Analysis

The QA process is crucial in order to analyse each question and build a relationship with the dataset correctly. Hence, a pre-processing phase is required, using NLP tasks to parse the main keywords into an ontology mapper. The main NLP tasks required are normalization, tokenization, removing the stop words, POS tagging, and stemming. In the following sections, we explain each task briefly.

5.3.1. Question Analysis Example

In this section, an example of the question analysis process is presented. We present only one question as an example, assuming that other questions have similar structure. Figure 5.3 presents the pre-processing stages for the asked question as below:

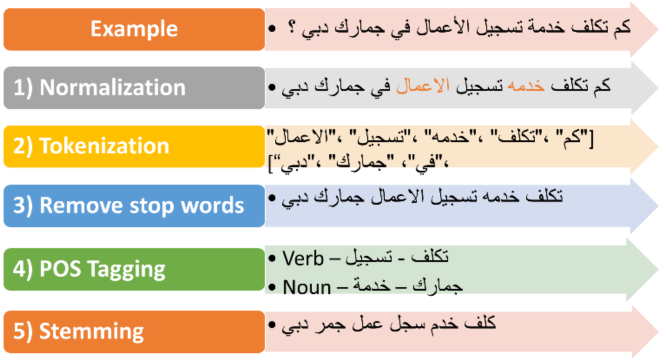

Example
• كم تكلف خدمة تسجيل الأعمال في جمارك دبي ؟
1) Normalization
• كم تكلف خدمه تسجيل الاعمال في جمارك دبي
2) Tokenization
["كم"، "تكلف"، "خدمه"، "تسجيل"، "الاعمال"،
"في"، "جمارك"، "دبي"]
3) Remove stop words
• تكلف خدمه تسجيل الاعمال جمارك دبي
4) POS Tagging
• Verb – تسجيل - تكلف
• Noun – خدمة – جمارك
5) Stemming
• كلف خدم سجل عمل جمر دبي

Figure 5.3: Question Pre-processing Example

- Normalization stage: Section 5.3.2 illustrates this stage, and we will follow it. The punctuation "؟" is removed, and the "ة" is replaced with "ه" in the word "خدمة", as well as the Alif "أ" with "ا" in the word "الأعمال". The sentence after normalization becomes "كم تكلف خدمه تسجيل الاعمال في جمارك دبي".
- Tokenization stage: The question is distributed into tokens

[“دبي"، "جمارك" ،"في"، "الاعمال"، "تسجيل"، "خدمه"، "تكلف"، "كم"]

- Remove stop words: The stop word "في" is removed.
- POS Tagging: In this stage, the POS tagging process produces the verbs "تكلف" and "تسجيل“ and the nouns "جمارك" and "خدمة".
- Stemming: In this stage, the following words are stemmed: "تكلف"becomes "كلف", "خدمه" becomes "خدم", "تسجيل" becomes "سجل", "الاعمال" becomes "عمل", "جمارك" becomes "جمر", and "دبي" becomes "دبي").

5.3.2. Normalization

The aim of this process is to convert the text into one canonical form to create consistency between the question and the ontology. The main steps for normalization according to Dilekh and Behloul (2012) are as follows:

- Remove punctuation, diacritics, and non-letters.
- Replace "أ" or "إ" or "آ"with alif "ا".
- Replace "ىء" with "ئ".
- Replace last "ى" with "ي".
- Replace last "ة" with "ه".

5.3.3. Tokenization

After normalizing the question, the tokenization process is performed to split the text into tokens (words, symbols, phrases, or any other elements). The tokenizer technique segments the text based on the spaces between words. For further explanation, the previous example is tokenized to produce the following tokens:

["دبي"، "جمارك" ،"في"، "الاعمال"، "تسجيل"، "خدمه"، "تكلف"، "كم"]

5.3.4. Stop words removal

In our approach, we focus on question keywords to find the correct answer from the dataset. Therefore, the stop words (prepositions, conjunctions, or words frequently used in the sentences) are removed. The list of stop words was captured from GitHub (2018). The percentage of deleted stop words from the service class is 6.08% and from the overall dataset is 13.01%.

5.3.5. Part-of-Speech Tagging

This task is important for question analysis. The objective is to tag the questions' verbs, nouns, adjectives, adverbs, etc. (El Hadj, Al-Sughayeir & Al-Ansari 2009). After POS tagging the question, we match the tagged verbs with the main keywords in the ontology classes. Accordingly, NL questions can easily be translated into SPARQL queries.

5.3.6. Stemming

Stemming tasks decrease a word to its root and retain it as a representational word. Stemming in Arabic is more complicated than in English. The amount of inflection in English is quite low, so the root is close to the actual word. Conversely, Arabic roots have different forms than the actual words after removing suffixes and prefixes (Riloff & Thelen 2000). For instance, the root is "نهى" for the following words: "تنتهي", "نهائي", "المنتهية". In this study, we use the ISRI Arabic Stemmer (Abainia, Ouamour & Sayoud 2017), which has the following steps to stem Arabic words:

1. Character normalisation
2. Prefix removal (for example, "بال, كال, وال, فال, وبال, فكال, فلل, ولل, ال, لل, فل, فب")
3. Suffix removal (for example, "ك, كي, كم, كما, كنّ, ه, ها, هم, هما, هنّ, نا")
4. Plural transformation to singular
5. Feminine transformation to masculine
6. Verb stemming

5.4. IR and Question-Ontology Mapping

This section is the core process for our model. The ontology mapping is implemented through two types of mapping: semantics-based and keyword-based. The semantics-based approach searches for the answer by analysing a user question to understand the main concepts required from the ontology, whereas the keyword-based approach gets the mapped keywords as a text query without analysing or understanding the ontological concepts. The approach of each is explained below with a supporting example.

5.4.1. Semantics-Based Mapping

The apparent purpose of semantics-based mapping is to find the answer to a user's question based on an ontology meta-model by mapping the question words with the ontology components. The process is established by mapping the question with ontology components after pre-processing for both sides. Each question has a set of extracted terms, and each word is mapped with ontology classes, data properties, and objects. If the mapping is built successfully, it can be applied to build the SPARQL query to retrieve the answer to the NL query.

Our ontology has three main classes ("S_الجهة_الحكومية", "S_الخدمة", and "Keywords"), and each word in an NL question has to be mapped with these classes according to the extracted keywords. Moreover, each instance has a set of data properties that are eligible for the mapping process. The keywords class contains all ontological keywords and maps to other classes. Consequently, a data object called "linked_to_keywords" is created to link the class with its keywords. After that, the instances that are related to the keywords are retrieved, but in order to retrieve the right answer, more rules are required to link the classes. "Link_to_Service" and "Link_to_Entity" object properties link the classes "S_الجهة_الحكومية" and "S_الخدمة" together. Hence, the relations between classes are trusted more to retrieve the right answer. We are using an N-gram algorithm to solve the challenge of matching the question terms with ontology keywords. The matching process is built based on the itemset. Figure 5.4 illustrates an example for the question–ontology mapping process.

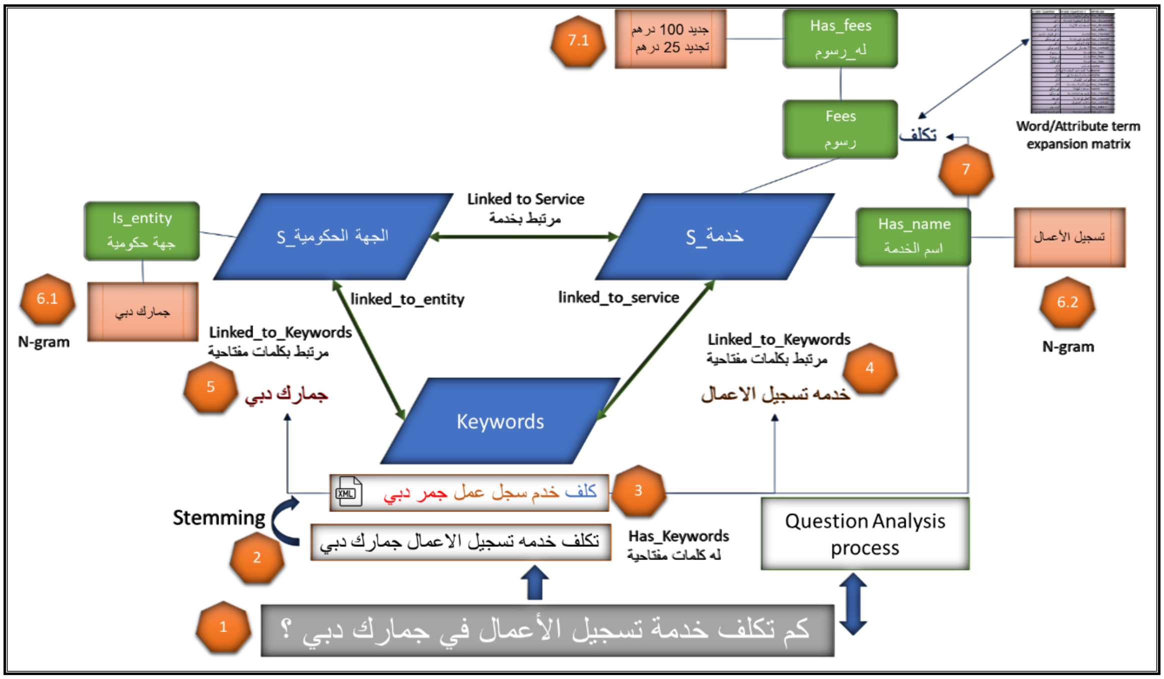


Figure 5.4: Question–Ontology Mapping Process Example

1. The NL question is “كم تكلف خدمة تسجيل الأعمال في جمارك دبي ؟”.
2. The keywords registered in the class Keywords are تكلف، خدمه، تسجيل، الاعمال، جمارك، دبي.
3. The stemming words and the keywords are registered in an XML repository in order to map the stemmed keywords (كلف، خدم، سجل، عمل، جمر، دبي) with the question keywords that have multiple forms with a single root.
4. The words خدمه، تسجيل، الاعمال are the keywords in the class S_الخدمة and linked to the object property Linked_to_keyword.
5. The words جمارك، دبي are the keywords in the class S_الجهة_الحكومية and linked to the object property Linked_to_keyword.
6. An N-gram is used to match the question with keywords for each class.
   6.1. The class CL_الجهة_الحكومية has the keywords جمارك ، دبي ; 2-gram matching retrieves Instance CL_جمارك_دبي , which has the data property is_entity.
   6.2. The class S_الخدمة has the keywords تسجيل، الاعمال; 2-gram matching retrieves Instance S_تسجيل_الأعمال , which is linked with class: S_خدمة.
7. After retrieving the service and entity name, we need to retrieve the data property. The word تكلف is mapped to the Datatype property Has_fees in the ontology. This mapping comes from the Word/Attribute term expansion matrix, as mentioned in Section 5.2.

7.1. The value of the Data Property Has_fees “جديد 100 درهم تجديد 25 درهم” is the answer to the user’s question.

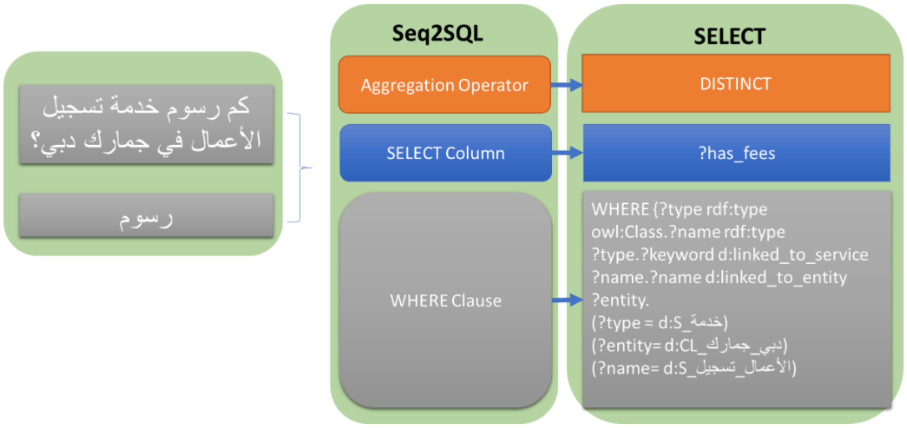
5.4.2. Keyword-Based Mapping

This approach gets the question’s answer by matching the question keywords in a text after stemming it with the ontology, without analysing the ontology concepts. Below is an example with steps to explain this approach:

1. The user asks the question " ما هي ارقام التواصل لخدمة طلب تحديد جلسة في محاكم دبي ؟".
2. Question pre-processing tasks (normalization, tokenization, POS tagging, removing stop words, and stemming) are performed to return the stemmed keywords. The output of this process is " رقم، خدم، طلب، حدد، جلس، محا، دبي ".
3. According to Section 5.2, the word التواصـل in the question is mapped with the Data Property has_url, which points to the address of the website that has the expected question answer.
4. The SPARQL query is formulated per Section 5.4.3, returning the first answer by adding the filters for all stemmed keywords.

5.4.3. SPARQL Query Generation

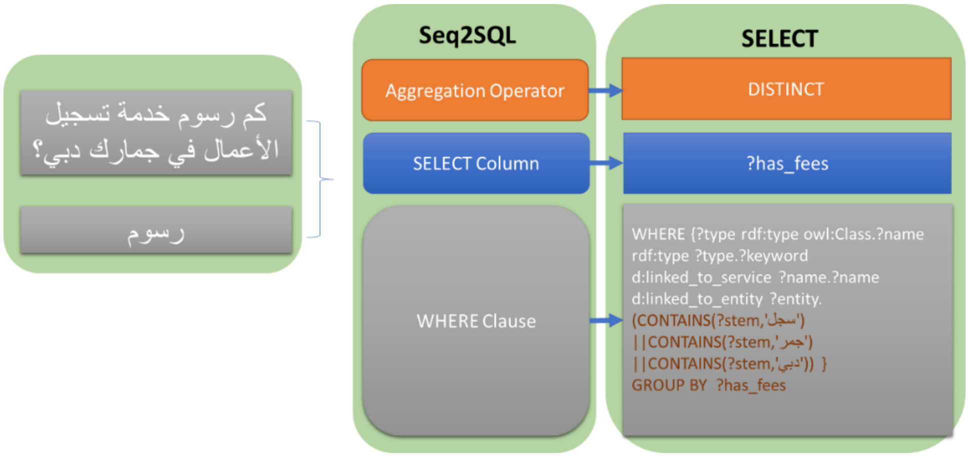
In this section, we need to translate the questions into SPARQL query format. We adopt Zhong, Xiong & Socher’s (2017) approach to generate the query. The Seq2SQL approach uses reinforcement learning by executing a loop-query over the dataset to understand a policy in order to generate the query. Further, this approach has three main parts to building the query: SELECT COLUMN, WHERE clause, and Aggregation Operator. (1) **SELECT Column** relies on the data properties of ontology instances. For example, "كم رسـوم خدمة تسـجيل الأعمال؟" indicates that the fees “رســوم” have to be retrieved. To generate the right columns from the ontology, we prepared in Section 5.2 a matrix mapping the ontology data properties with the main keywords of questions. (2) **WHERE Clause** is the core process of SPARQL query, used to indicate the main ontology classes that are required in the query based on question analysis. Also, the relation rules are generated in this part to link the ontology classes together. For instance, the previous example required mapping the classes “S_الجهة_الحكومية”, “S_الخدمة”, and “Keywords” by adding the rules “linked_to_service”, “linked_to_entity”, and “is_entity”. In addition, there are standard rules applied to all generated queries, such as "?type rdf:type owl:Class." and "?name rdf:type ?type.". (3) The **Aggregation Operator** depends on the question. For instance, Count, Sum, Group By, Distinct, etc. can be used Figure 5.5 presents an example of a semantics-based approach translated to SPARQL Query.



```
WHERE {?type rdf:type
owl:Class.?name rdf:type
?type.?keyword d:linked_to_service
?name.?name d:linked_to_entity
?entity.
(?type = d:S_خدمة)
(?entity= d:CL_دبي_جمارك)
(?name= d:S_الأعمال_تسجيل)
```

Figure 5.5: Semantics-based SPARQL Example

We adopt the same algorithm in the keyword-based approach to building the SPARQL query. The differences between the semantics-based and keyword-based approaches are located in the WHERE clause, so the ontology classes are eliminated in the keyword-based approach, and the stemming words from the questions are added to retrieve the answer from the whole ontology. Figure 5.6 illustrates an example of keyword-based SPARQL structure.



```
WHERE {?type rdf:type owl:Class.?name
rdf:type ?type.?keyword
d:linked_to_service ?name.?name
d:linked_to_entity ?entity.
(CONTAINS(?stem,'سجل')
||CONTAINS(?stem,'جمر')
||CONTAINS(?stem,'دبي')) }
GROUP BY ?has_fees
```

Figure 5.6: Keyword-based SPARQL Example

5.5. Answer Analysis

The answer analysis process has two main parts: answer extraction and validation. Answer extraction is done by SPARQL queries, as in Section 5.4.3. Then, domain experts validate the answer candidates by tagging each answer as correct or not depending on the ontology and the defined semantic rules. The domain experts check whether the top answer matches the exact answer.

5.6. Challenges

While generating the SPARQL queries, some limitations and issues occurred, which we can summarize as follows:

- **Missing ontology instances**: The algorithm translated the semantics-based queries correctly, but there are missing ontology instances in the query because of missing identified rules mapping between the ontology data properties and questions’ keywords. For instance, in the question " ما هي وســائل تقديم الخدمة طلب ملكية مركبة بدل فاقد تالف ", the word "وسائل" is not linked with the data type in the RDF structure. Therefore, we have associated this word in the Word/Attribute term expansion matrix, as mentioned in Section 5.2.
- **Limitation in stemming tool**: In the keyword-based approach, we rely on NLP techniques to extract the keywords from the question and map them with ontology keywords. Actually, all SPARQL queries were built successfully, but 114 questions did not return an answer due to the limitation on stemmed keywords. For instance, the question “ ما رسوم خدمة تحميل صـــور عن إمارة دبي في دائرة الســـياحة” did not return any answer. Its stemmed keywords are حمل, صور, مرة, دبي, دئر, سيح; words that can be stems for multiple keywords do not exist in our ontology. To resolve this issue, the ontology keywords repository has to be supported with more terms by using Arabic WordNet (AWN).

5.7. Experimental Results

We performed experimental tests to evaluate our algorithm’s enhancements of semantics-based and keyword-based approaches. The aim of choosing a keyword-based approach was to measure the keywords’ mapping with RDF instances, while the objective of the semantics-based approach was to get the correct answer from RDF instances. The QA algorithm was applied to the constructed ontology. To measure the quality of our algorithm and the relations of the constructed ontology, two experimental tests were conducted on 414 questions using semantics-based and keyword-based approaches. In addition, we compared our algorithm with the Google search engine by running the same number of questions.

The confusion matrix method was adopted to measure QA performance. The confusion matrix elements are illustrated below in Table 5.5.

	Retrieved	**Not Retrieved**
Relevant	**True Positive (TP)** Number of questions that are correctly answered by the algorithm	**False Positive (FP)** Number of questions that are answered by the algorithm but not correct.
Irrelevant	**False Negative (FN)** Number of questions that are not correct but retrieved by the algorithm.	**True Negative (TN)** Number of questions that are not correct and not retrieved by the algorithm

Table 5.5: QA Confusion Matrix

Precision, recall, and F-measure are the main metrics of the confusion matrix and are calculated based on the following formulas in Figure 5.7:

$$Precision = \frac{TP}{(TP + FP)} \qquad Recall = \frac{TP}{(TP + FN)} \qquad F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

$$Accuracy = \frac{TP + TN}{(TP + TN + FP + FP)}$$

Figure 5.7: QA Performance Measurement Formulas

Basically, the test questions are executed for both approaches, and the result shows high precision in the semantics-based approach with 95%, whereas the precision in the keyword-based approach achieves only 72%. In addition, the recall in the keyword-based approach achieves 100%, which indicates a high reliability level, while the recall in the semantics-based approach reaches 94%. The accuracy results for the semantics-based and keyword-based approaches are 90% and 72%, respectively. Low accuracy traced to the keyword repository requires more keywords to be extracted from the ontology and more terms for each keyword to be extracted from AWN. Furthermore, for quality purposes, 30 questions were submitted by domain experts to measure the performance of our approach. Keyword-based methods were used for these types of questions. The results for precision, recall, F1-measure, and accuracy are 67%, 100%, 80%, and 67%, respectively.

Finally, we compared our results with the well-known search engine Google. We manually submitted 414 questions on the Google search engine and got the top two results, then compared these results with the answers from our experimental test. The results are 55%, 100%, 71%, and 55% for precision, recall, F1-measure, and accuracy, respectively. Below, Table 5.6 summarises the experimental results.

Measurement Type	**Semantics-based**	**Keyword-Based**	**Users Questions**	**Google Search Engine**
# of Questions	414	414	30	414
Precision	95%	72%	67%	55%
Recall	94%	100%	100%	100%
F1-Measure	94%	84%	80%	71%
Accuracy	90%	72%	67%	55%

Table 5.6: QA Experimental Results

5.8. Summary

In this chapter, we presented our Arabic semantics-based QA model for the ontology constructed in Chapter Four. We developed a methodology to answer user questions in NL relevant to Dubai services. The methodology has three phases: question analysis, IR, and answer validation. We automatically built 414 questions related to the constructed ontology, divided into factoid, list, and complex questions. Each question was analysed using NLP techniques, then mapped to the ontology in two approaches: semantics-based and keyword-based. A SPARQL query was generated based on Zhong, Xiong & Socher's (2017) approach in order to extract the answer from the ontology. The challenges that occurred were explained and discussed. The experimental results show good results in precision, recall, F1-measure, and accuracy. Also, a comparison was conducted with the Google search engine that shows our accuracy to be better than Google's.

6. Chapter Six: Research Questions Answers

In this chapter, the research questions' answers are provided, with reference to the corresponding sections in the dissertation.

6.1. Research Question 1

Question: Is it possible to build a QA system that can answer government-related questions as accurately as possible, deal with Arabic QA challenges, and reach acceptable performance?

Answer: Yes, according to Chapter Four and Sections 5.2, 5.6, and 5.7, it was possible to answer government-related questions correctly by implementing a closed-domain ontology from the Dubai government's website portal. Section 5.2 presented the baseline questions that were used and their relevance to the government service catalogue. Also, the Arabic QA challenges were presented and resolved in section 5.6. Further, the QA model was implemented according to this ontology and achieved 90% accuracy. We compared our model with the Google search engine, and we achieved better performance results, as stated in Section 5.7.

6.2. Research Question 2

Question: How can we construct an ontology related to e-government services and enable information retrieval in a structured ontology?

Answer: In Chapter Four, we explained how we constructed an SW ontology for e-government services. Then, we proposed semantics-based mapping rules in Section 5.4 in order to retrieve the answer to the asked question from the structured ontology. Actually, the ontology was built according to the defined rules to create relations between all ontology components. Therefore, the IR was successfully applied to the structured ontology. Sections 4.6.1 and 4.7 illustrated the ontology construction results, and Section 5.7 illustrated the experimental results for IR.

6.3. Research Question 3

Question: How can we enhance the QAS based on a rule-based approach?

Answer: Generally, our QA model was developed based on semantics-based rules, which were defined in the ontology dataset. In Chapter Four, we built QA models based on two approaches: first, a semantics-based approach that depends on ontological rules; second, a keyword-based approach that relies on questions' analysis and on ontology keywords and terms. A comparison was conducted between the two approaches, and Section 5.7 proved that the rule-based approach enhanced the QA systems by adding more rules to the ontology. In addition, another comparison was conducted with the Google search engine, which confirmed that the rule-based approach has better results and improves the performance of QAs.

7. Chapter Seven: Conclusion, Limitations, and Future Prospects

This chapter summarises the work that have been done to achieve the objectives and answers the research questions of this dissertation. Moreover, it discusses the various limitations and we could minimize its impact. Lastly, we present future research directions in this domain specific area.

7.1. Conclusion

The main purpose of this study was to automatically build an ontology for e-government services in the UAE from the dubai.ae website portal. What's more, it aimed to apply a QA system to the constructed ontology. We proposed an approach to constructing an Arabic ontology for Dubai's e-government customer services. Dubai government services data sources were selected for this study that were built based on international standards. In our approach, all Arabic services' data were extracted from the dubai.ae portal using the AEDS tool, which we developed. An analysis was performed on all collected webpages to determine the main entities and attributes in order to make mapping rules between the OWL ontology components and the extracted keywords in XML form. Our methodology has two phases, automatic ontology construction and the Arabic question answering system. The first phase starts with the data extraction and validation process by using inter-annotator agreement, then measuring the quality of the dataset, and finally using NLP tasks to extract the terms/keywords and cleaning the dataset. OWL/RDF representation is the base of any ontology using the Protégé tool; therefore, OWL analysis was conducted to automatically generate the codes for header, class, property, and individual and to gather all codes in one repository that could be opened by Protégé. The experimental results show significant performance with high precision and recall for the extracted concepts and relations.

In the second phase, we developed an Arabic QA model depending on the constructed ontology to respond to questions related to e-government services. The proposed QA model contains three parts. Firstly, question analysis aimed to analyse the identified baseline questions as well as create correct rules for the dataset. Therefore, a pre-processing phase was mandated through conducting NLP tasks to parse the keywords into an ontology mapper engine. The main NLP tasks used were normalization, tokenization, removing the stop words, POS tagging, and stemming. Secondly, IR question–ontology mapping aimed to map the analysed question to our ontology in order to extract the correct answer. Actually, we used two approaches, semantics-based and keyword-based, to extract the answer, and we performed a comparison between them to confirm that the rules-based approach achieved better results than normal approaches. Four hundred fourteen questions were translated from NL to SPARQL queries to retrieve their answers from the constructed ontology and executed in the Jena framework. Also, 30 experts' questions were captured and tested by our QA model. The results of QA show higher performance results for the semantics-based than for the keyword-based approach.

In this dissertation, we have come across various challenges in constructing an ontology and a QAS. Extracting the correct keywords from all ontology components impacts ontology rules; therefore, we extracted the keywords from the service class using a statistical method that considered the base of ontology rules. In addition, extracting the services' profiles from Arabic webpages in an organized dataset containing attributes and relations created a challenge for us. This was resolved by developing the AEDS tool to extract the dataset correctly. Moreover, there was a challenge in the POS tagger tool, which tagged some verbs as nouns; this challenge was resolved manually in a post-processing step. In the QA methodology, there were some challenges. First, there were missing ontology instances when executing SPARQL queries due to missing identified rules mapping between the ontology data properties and questions' keywords. This was resolved by mapping the question types to ontology data attributes. Further, there were some queries executed without returning any answer due to limitations on the stemming tool, which was partially resolved by supporting ontology keywords with terms from AWN.

The main contribution of this dissertation is automatically building an ontology from Arabic webpages in the OWL format. Further, the XML schema was used for ontology keywords to bridge the gaps between the main entities' terms and ontology components. Another key thing is that linguistic processing was adopted to maintain the SW components, which is important for building components' rules. Finally, the QA model was built for this ontology to answer Arabic user questions correctly. Semantic analysis, NLP, and SPARQL were used to formulate the user queries based on keywords extracted from the ontology and NL queries.

7.2. Limitations

In the ontology construction phase, we tried to design more concepts to cover all services' details mentioned in the government service catalogue. However, we were not able to obtain all details from one source due to there being different data sources. In addition, we did not have enough time to build an automatic approach to validating the ontology's quality; therefore, we relied on domain experts for ontology construction validation. In the QAS phase, we did not have time to implement a tool to compare the questions' results with the Google search engine automatically. Therefore, we used a semi-manual approach in the evaluation stage.

7.3. Future Prospects

In this section, some suggestions are presented for future works. Regarding ontology construction, there is an opportunity for future work to extend the dataset with more details that contain enough information for all customers or government partnerships from one central entity. This will be a strong base on which to build a solid dataset including sufficient and up-to-date information. Furthermore, it is possible to extract and validate the Arabic keywords using AWN as well as more reliable and well-known techniques such as term frequency/inverted document frequency (TF/IDF), text mining techniques, and support vector machines (SVM).

Regarding the question answering, we are looking forward to developing a system with a complete QA process that will be used by government entities and increasing the accuracy of QA. We suggest using similarity measurements for OWL ontology components to define the keywords and terms.

References

Abainia, K., Ouamour, S. & Sayoud, H. (2017). A novel robust Arabic light stemmer. *Journal of Experimental and Theoretical Artificial Intelligence*.

Abdelali, A., Darwish, K., Durrani, N. & Mubarak, H. (2016). Farasa: A fast and furious segmenter for arabic. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pp. 11–16.

Abouenour, L., Bouzoubaa, K. & Rosso, P. (2008). Improving Q/A using Arabic wordnet. *Proc. The 2008 International Arab Conference on Information Technology (ACIT'2008), Tunisia, December*.

AbuTaha, A. W. & Alagha, I. M. (2015). An Ontology-Based Arabic Question Answering System. *Central library of Islamic University of Ghaza*.

Al-Chalabi, H., Ray, S. & Shaalan, K. (2015). Semantic Based Query Expansion for Arabic Question Answering Systems. *Arabic Computational Linguistics (ACLing), 2015 First International Conference on*, pp. 127–132.

AL-Khawaldeh, F. T. (2015). Answer Extraction for Why Arabic Questions Answering Systems: EWAQ. *World of Computer Science & Information Technology Journal*, vol. 5(5).

Al-Safadi, L., Al-Badrani, M. & Al-Junidey, M. (2011). Developing ontology for Arabic blogs retrieval. *International Journal of Computer Applications*. International Journal of Computer Applications, 244 5 th Avenue,# 1526, New York, NY 10001, USA India, vol. 19(4), pp. 40–45.

Al-Shalabi, R., Kanaan, G., Al-Sarayreh, B., Khanfar, K., Al-Ghonmein, A., Talhouni, H. & Al-Azazmeh, S. (2009). Proper noun extracting algorithm for arabic language. *International Conference on IT to Celebrate S. Charmonman's 72nd Birthday*, pp. 21–28.

Al-Zoghby, A. M. & Shaalan, K. (2015a). Conceptual search for Arabic web content. *International Conference on Intelligent Text Processing and Computational Linguistics*, pp. 405–416.

Al-Zoghby, A. M. & Shaalan, K. (2015b). Semantic Search for Arabic. *FLAIRS Conference*, pp. 524–529.

Alazemi, N. N., Al-Shehab, A. J. & Alhakem, H. A. (2017). E-government Frameworks based on Semantic Web Services: A Comprehensive Study. *International Journal of Computer Applications*. Foundation of Computer Science, vol. 158(7).

Albarghothi, A., Khater, F. & Shaalan, K. (2017). Arabic Question Answering Using Ontology. *Procedia Computer Science*. Elsevier, vol. 117, pp. 183–191.

Allam, A. M. N. & Haggag, M. H. (2012). The question answering systems: A survey. *International Journal of Research and Reviews in Information Sciences (IJRRIS)*, vol. 2(3).

Alqaryouti, O., Siyam, N. & Shaalan, K. (2019). 'A sentiment analysis lexical resource and dataset for government smart apps domain'. , in *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2018*. Springer, Cham, pp. 230–240.

Al Ameed, H., Al Ketbi, S., Al Kaabi, A., Al Shebli, K., Al Shamsi, N., Al Nuaimi, N. & Al

Muhairi, S. (2005). Arabic light stemmer: A new enhanced approach. *The Second International Conference on Innovations in Information Technology (IIT'05)*, pp. 1–9.

Arabic WordNet. (2018) [online]. [Accessed 15 April 2018]. Available at: http://globalwordnet.org/arabic-wordnet/.

Bekhti, S. & Alharbi, M. (2013). AQuASys: A question-answering system for Arabic. *Proceedings of WSeas International Conference. Recent Advances in Computer Engineering Series*.

Bekkers, V. (2003). E-government and the emergence of virtual organizations in the public sector. *Information Polity: an international journal on the development, adoption, use and effects of information technology*.

Benabdallah, A., Abderrahim, M. A. & Abderrahim, M. E.-A. (2017). Extraction of terms and semantic relationships from Arabic texts for automatic construction of an ontology. *International Journal of Speech Technology*. Springer, vol. 20(2), pp. 289–296.

Benajiba, Y., Rosso, P., Abouenour, L., Trigui, O., Bouzoubaa, K. & Belguith, L. (2014). 'Question answering'. , in *Natural Language Processing of Semitic Languages*. Springer, pp. 335–370.

Benajiba, Y., Rosso, P. & Lyhyaoui, A. (2007). Implementation of the ArabiQA question answering system's components. *Proc. Workshop on Arabic Natural Language Processing, 2nd Information Communication Technologies Int. Symposium, ICTIS-2007, Fez, Morroco, April*, pp. 3–5.

Berners-Lee, T. & Fischetti, M. (2001). *Weaving the Web: The original design and ultimate destiny of the World Wide Web by its inventor*. DIANE Publishing Company.

Boyce, S. & Pahl, C. (2007). Developing domain ontologies for course content.

Brank, J., Grobelnik, M. & Mladenić, D. (2005). A survey of ontology evaluation techniques.

Brini, W., Ellouze, M., Trigui, O., Mesfar, S., Belguith, H. L. & Rosso, P. (2009). Factoid and definitional Arabic question answering system. *Post-Proc. NOOJ-2009, Tozeur, Tunisia, June*, pp. 8–10.

Chang, C.-H., Kayed, M., Girgis, M. R. & Shaalan, K. F. (2006). A survey of web information extraction systems. *IEEE transactions on knowledge and data engineering*. IEEE, vol. 18(10), pp. 1411–1428.

Charalabidis, Y. & Metaxiotis, K. (2009). 'Ontology-based management of e-government knowledge'. , in *Social and Political Implications of Data Mining: Knowledge Management in E-Government*. IGI Global, pp. 221–234.

Clarke, E. L., Loguercio, S., Good, B. M. & Su, A. I. (2013). A task-based approach for Gene Ontology evaluation. *Journal of biomedical semantics*, p. S4.

d'Aquin, M. & Motta, E. (2011). Extracting relevant questions to an RDF dataset using formal concept analysis. *Proceedings of the sixth international conference on Knowledge capture*, pp. 121–128.

Damljanovic, D., Agatonovic, M. & Cunningham, H. (2010). Natural language interfaces to ontologies: Combining syntactic analysis and ontology-based lookup through the user interaction. *Extended Semantic Web Conference,* pp. 106–120.

Dang, H. T., Kelly, D. & Lin, J. J. (2007). Overview of the TREC 2007 Question Answering Track. *Trec,* p. 63.

Darwish, K. & Mubarak, H. (2016). Farasa: A New Fast and Accurate Arabic Word Segmenter. *LREC*.

Dellschaft, K. & Staab, S. (2006). On how to perform a gold standard based evaluation of ontology learning. *International Semantic Web Conference,* pp. 228–241.

Dhanjal, G. S. & Sharma, S. (2015). Advancements in Question Answering Systems Towards Indic Languages. *International Journal of Research in Computer Science*, vol. 5(1), pp. 15–26.

Dilekh, T. & Behloul, A. (2012). Implementation of a new hybrid method for stemming of Arabic text. *analysis,* vol. 3(4), p. 5.

Dostal, M. & Ježek, K. (2011). Automatic keyphrase extraction based on NLP and statistical method. Západočeská univerzita v Plzni.

Dubai Government. (2018). *Dubai Government* [online]. [Accessed 26 January 2018]. Available at: http://www.dubai.ae.

Ezzeldin, A. M. & Shaheen, M. (2012). A survey of Arabic question answering: challenges, tasks, approaches, tools, and future trends. *Proceedings of The 13th International Arab Conference on Information Technology (ACIT 2012),* pp. 1–8.

Farghaly, A. & Shaalan, K. (2009). Arabic natural language processing: Challenges and solutions. *ACM Transactions on Asian Language Information Processing (TALIP)*. ACM, vol. 8(4), p. 14.

Ferdinand, M., Zirpins, C. & Trastour, D. (2004). Lifting XML schema to OWL. *International conference on web engineering*, pp. 354–358.

Gangwal, G. (2012). Question Answering System using Open Source Software.

Gil-Garcia, J. R. & Martinez-Moyano, I. J. (2007). Understanding the evolution of e-government: The influence of systems of rules on public sector dynamics. *Government Information Quarterly*. Elsevier, vol. 24(2), pp. 266–290.

Gilmmour, R. (2004). An ontology for hazard identification in risk management.

GitHub. (2018). *Alir3z4/python-stop-words.* [online]. [Accessed 31 January 2018]. Available at: https://github.com/Alir3z4/python-stop-words.

Google Search Statistics - Internet Live Stats. (n.d.) [online]. [Accessed 5 February 2018]. Available at: http://www.internetlivestats.com/google-search-statistics/.

Gorenjak, B., Ferme, M. & Ojsteršek, M. (2011). A question answering system on domain specific knowledge with semantic web support. *International journal of computers*, vol. 5(2), pp. 141–148.

Guijarro, L. (2009). Semantic interoperability in eGovernment initiatives. *Computer*

Standards & Interfaces. Elsevier, vol. 31(1), pp. 174–180.

GulfNews. (2015). *New Dubai meter to gauge happiness | GulfNews.com* [online]. [Accessed 18 February 2018]. Available at: https://gulfnews.com/news/uae/government/new-dubai-meter-to-gauge-happiness-1.1496207.

Gupta, P. & Gupta, V. (2012). A survey of text question answering techniques. *International Journal of Computer Applications*. Foundation of Computer Science, vol. 53(4).

Gyawali, B. (2011). Answering Factoid Questions via Ontologies: A Natural Language Generation Approach. *University of Malta*.

Haarmann, B., Gottsmann, F. & Schade, U. (2012). How to make Ontologies self-building from Wiki-Texts. *Proceedings of the International Conference on Information and Knowledge Engineering (IKE)*, p. 1.

El Hadj, Y., Al-Sughayeir, I. & Al-Ansari, A. (2009). Arabic part-of-speech tagging using the sentence structure. *Proceedings of the Second International Conference on Arabic Language Resources and Tools, Cairo, Egypt*.

Hamdy, M. E. & Shaalan, K. (2018). A Hybrid Framework for Applying Semantic Integration Technologies to Improve Data Quality. *International Journal of Information Technology and Language Studies*, vol. 2(1).

Hammo, B., Abu-Salem, H. & Lytinen, S. (2002). QARAB: A question answering system to support the Arabic language. *Proceedings of the ACL-02 workshop on Computational approaches to semitic languages*, pp. 1–11.

Hazman, M., El-Beltagy, S. R. & Rafea, A. (2012). An ontology based approach for automatically annotating document segments. *IJCSI International Journal of Computer Science Issues*, vol. 9(2), pp. 221–230.

Hebeler, J., Fisher, M., Blace, R. & Perez-Lopez, A. (2009). *Semantic Web programming*. Indianapolis, Indiana: Wiley.

Hitzler, P., Krotzsch, M. & Rudolph, S. (2009). *Foundations of semantic web technologies*. Chapman and Hall/CRC.

Imam, I., Nounou, N., Hamouda, A. & Khalek, H. A. A. (2013). An ontology-based summarization system for arabic documents (ossad). *Int. J. Comput. Appl*, vol. 74(17), pp. 38–43.

Jain, V. & Singh, M. (2013). Ontology development and query retrieval using protégé tool. *International Journal of Intelligent Systems and Applications*, vol. 9, pp. 67–75.

Jarrar, M. (2010). The Arabic Ontology. *Lecture Notes, Knowledge Engineering Course (SCOM7348), Birzeit University, Palestine*.

Jurafsky, D. & Martin, J. (2008). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*.

Khaleej Times. (2013). *Dubai School of Government renamed after Mohammed - Khaleej Times* [online]. [Accessed 15 February 2018]. Available at: https://www.khaleejtimes.com/article/20130701/ARTICLE/307019969/1014.

Khoja, S. (2001). APT: Arabic part-of-speech tagger. *Proceedings of the Student Workshop at NAACL*, pp. 20–25.

Kurdi, H., Alkhaider, S. & Alfaifi, N. (2014). Development and evaluation of a web based question answering system for Arabic language. *Computer Science & Information Technology (CS & IT)*, vol. 4(02), pp. 187–202.

Laboratory for Data Technologies. (n.d.). *Laboratory for Data Technologies - Semantic web and Ontologies* [online]. [Accessed 22 March 2018]. Available at: http://lpt.fri.uni-lj.si/research/15-semantic-web-and-ontologies/6-semantic-web-and-ontologies.

Lahbari, I., El Alaoui, S. O. & Zidani, K. A. (2018). Toward a new arabic question answering system. *Int. Arab J. Inf. Technol.*, vol. 15(3A), pp. 610–619.

Lampert, A. (2004). A quick introduction to question answering. *Dated December*.

Lim, E.-P. & Sun, A. (2005). Web mining-The ontology approach. *Proceedings of The International Advanced Digital Library Conference (IADLC 2005), Nagoya, Japan (August 2005)*.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pp. 55–60.

McCarthy, L., Vandervalk, B. & Wilkinson, M. (2012). SPARQL Assist language-neutral query composer. *BMC bioinformatics*. BioMed Central, vol. 13(1), p. S2.

McHugh, M. L. (2012). Interrater reliability: the kappa statistic. *Biochemia medica: Biochemia medica*. Medicinska naklada, vol. 22(3), pp. 276–282.

Mehdipour Pirbazari, A. (2017). *Answering Engine for sports statistics: Development of an ontology and a knowledge base*. University of Stavanger, Norway.

Mihalcea, R., Liu, H. & Lieberman, H. (2006). NLP (natural language processing) for NLP (natural language programming). *International Conference on Intelligent Text Processing and Computational Linguistics*, pp. 319–330.

Mishra, A. & Jain, S. K. (2016). A survey on question answering systems with classification. *Journal of King Saud University-Computer and Information Sciences*. Elsevier, vol. 28(3), pp. 345–361.

Mislove, A., Marcon, M., Gummadi, K. P., Druschel, P. & Bhattacharjee, B. (2007). Measurement and analysis of online social networks. *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement - IMC '07*. New York, New York, USA: ACM Press, p. 29.

Mohammed, F. A., Nasser, K. & Harb, H. M. (1993). A knowledge based Arabic question answering system (AQAS). *ACM SIGART Bulletin*. ACM, vol. 4(4), pp. 21–30.

Moldovan, D., Pa\csca, M., Harabagiu, S. & Surdeanu, M. (2003). Performance issues and error analysis in an open-domain question answering system. *ACM Transactions on Information Systems (TOIS)*. ACM, vol. 21(2), pp. 133–154.

Noy, N. F., McGuinness, D. L. & others. (2001). Ontology development 101: A guide to creating your first ontology. Stanford knowledge systems laboratory technical report KSL-01-05 and Stanford medical informatics technical report SMI-2001-0880, Stanford, CA.

Obrst, L., Werner, C., Inderjeet, M., Steve, R. & Smith, B. (2007). The Evaluation of Ontologies: Toward Improved Semantic Interoperability. Dans JO Christopher, Baker, & K.-H. Cheung. *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*.

Ou, S., Orasan, C., Mekhaldi, D. & Hasler, L. (2008). Automatic Question Pattern Generation for Ontology-based Question Answering. *Flairs Conference*, pp. 183–188.

Patel-Schneider, P. F. & Horrocks, I. (2006). *OWL 1.1 Web Ontology Language*. World Wide Web Consortium [online]. [Accessed 22 July 2018]. Available at: http://www.w3.org/Submission/2006/SUBM-owl11-overview-20061219/.

Protege Wiki. (2010) [online]. [Accessed 22 April 2018]. Available at: https://protegewiki.stanford.edu/wiki/Main_Page.

Quaresma, P. & Rodrigues, I. (2005). A logic programming based approach to QA@ CLEF05 track. *Workshop of the Cross-Language Evaluation Forum for European Languages*, pp. 351–360.

Ray, S. K. & Shaalan, K. (2016). A review and future perspectives of arabic question answering systems. *IEEE Transactions on Knowledge and Data Engineering*. IEEE, vol. 28(12), pp. 3169–3190.

Riloff, E. & Thelen, M. (2000). A rule-based question answering system for reading comprehension tests. *ANLP/NAACL 2000 Workshop on Reading comprehension tests as evaluation for computer-based language understanding sytems* -.

Rodrigues, T., Rosa, P. & Cardoso, J. (2006). Mapping XML to Exiting OWL ontologies. *International Conference WWW/Internet*, pp. 72–77.

Rosso, P., Benajiba, Y. & Lyhyaoui, A. (2006). Towards an Arabic question answering system. *Proc. 4th Conf. on Scientific Research Outlook & Technology Development in the Arab world, SROIV, Damascus, Syria*, pp. 11–14.

Rubin, D. L., Noy, N. F. & Musen, M. A. (2007). Protege: a tool for managing and using terminology in radiology applications. *Journal of digital imaging*. Springer, vol. 20(1), pp. 34–46.

Schwarzer, M., Düver, J., Ploch, D. & Lommatzsch, A. (2016). An Interactive e-Government Question Answering System. *LWDA*, pp. 74–82.

Shah, H., Shah, P. & Deulkar, K. (2015). A survey of ontology evaluation techniques for data retrieval. *International Journal Of Engineering And Computer Science*, vol. 4(11).

Sonbol, R., Ghneim, N. & Desouki, M. S. (2009). An Application Oriented Arabic Morphological Analyzer. *Proceedings of the workshop of morphological analyzer experts for Arabic language, organized by Arab League Educational, Cultural and Scientific Organization (ALECSO), King Abdul-Aziz City of Science and Technology (KACST) and Arabic Language Academy*.

Sosnovsky, S. & Dicheva, D. (2010). Ontological technologies for user modelling. *International Journal of Metadata, Semantics and Ontologies*. Inderscience Publishers, vol. 5(1), pp. 32–71.

Stanford Center. (2016). *Stanford Center* [online]. [Accessed 25 February 2018]. Available at: https://protege.stanford.edu/.

Statista. (2018). • *U.S. search engines: number of core searches 2018 | Statistic* [online]. [Accessed 5 February 2018]. Available at: https://www.statista.com/statistics/265796/us-search-engines-ranked-by-number-of-core-searches/.

Stoyanchev, S., Song, Y. C. & Lahti, W. (2008). Exact phrases in information retrieval for question answering. *Coling 2008: Proceedings of the 2nd workshop on Information Retrieval for Question Answering*, pp. 9–16.

Takala, P., Malo, P., Sinha, A. & Ahlgren, O. (2014). Gold-standard for Topic-specific Sentiment Analysis of Economic Texts. *LREC*, pp. 2152–2157.

Tatu, M., Balakrishna, M., Werner, S., Erekhinskaya, T. & Moldovan, D. (2016). Automatic extraction of actionable knowledge. *Semantic Computing (ICSC), 2016 IEEE Tenth International Conference On*, pp. 396–399.

The Executive Council of Dubai. (2015) [online]. [Accessed 26 January 2018]. Available at: https://tec.gov.ae.

W3C. (2015). *W3C Semantic Web Activity*. *W3C* [online]. [Accessed 22 April 2018]. Available at: https://www.w3.org/2001/sw/.

Waller, V. (2016). Making knowledge machine-processable: some implications of general semantic search. *Behaviour & Information Technology*. Taylor & Francis, vol. 35(10), pp. 784–795.

World Wide Web Consortium. (2012). *OWL 2 Web Ontology Language Document Overview (Second Edition)* [online]. [Accessed 25 March 2018]. Available at: https://www.w3.org/TR/owl2-overview/.

Xu, J. & Li, W. (2007). Using relational database to build OWL ontology from XML data sources. *Computational Intelligence and Security Workshops, 2007. CISW 2007. International Conference on*, pp. 124–127.

Yadav, U., Narula, G. S., Duhan, N., Jain, V. & Murthy, B. K. (2016). Development and visualization of domain specific ontology using protege. *Indian Journal of Science and Technology*, vol. 9(16), pp. 1–7.

Zhong, V., Xiong, C. & Socher, R. (2017). Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning [online]. [Accessed 16 July 2018]. Available at: http://arxiv.org/abs/1709.00103.